

SPIRAL: A superlinearly convergent incremental proximal algorithm for nonconvex finite sum minimization*

Pourya Behmandpoor[†] Puya Latafat[†] Andreas Themelis[‡] Marc Moonen[†]
Panagiotis Patrinos[†]

Abstract

We introduce SPIRAL, a **Su**Perlinearly convergent **In**cremental **pR**oximal **AL**gorithm, for solving nonconvex regularized finite sum problems under a relative smoothness assumption. Each iteration of SPIRAL consists of an inner and an outer loop. It combines incremental gradient updates with a linesearch that has the remarkable property of never being triggered asymptotically, leading to superlinear convergence under mild assumptions at the limit point. Simulation results with L-BFGS directions on different convex, nonconvex, and non-Lipschitz differentiable problems show that our algorithm, as well as its adaptive variant, are competitive to the state of the art.

Keywords Finite sum minimization, nonsmooth nonconvex optimization, relative smoothness, superlinear convergence, KL inequality

Mathematics Subject Classification (2000) 90C06, 90C25, 90C26, 49J52, 49J53, 90C53

1 Introduction

We study nonconvex nonsmooth finite sum optimization problems of the form:

$$\text{minimize}_{z \in \mathbb{R}^n} \varphi(z) := f(z) + g(z), \quad \text{where } f(z) := \frac{1}{N} \sum_{i=1}^N f_i(z). \quad (1.1)$$

The following basic assumptions are considered throughout the paper:

Assumption 1 (basic assumptions).

A1 $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ is L_i -smooth relative to a distance-generating function $h_i : \mathbb{R}^n \rightarrow \mathbb{R}$ (cf. [Definitions 2.1 and 2.3](#)), $i \in [N] := \{1, \dots, N\}$;

A2 $g : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ is proper and lower semicontinuous (lsc);

A3 a solution exists: $\arg \min \varphi \neq \emptyset$.

*P. Behmandpoor and M. Moonen acknowledge the research work carried out at the ESAT Laboratory of KU Leuven, in the frame of Research Project FWO nr. G0C0623N 'User-centric distributed signal processing algorithms for next generation cell-free massive MIMO based wireless communication networks' and Fonds de la Recherche Scientifique - FNRS and Fonds voor Wetenschappelijk Onderzoek - Vlaanderen EOS Project no 30452698 '(MUSE-WINET) Multi-Service Wireless NETworks'. The scientific responsibility is assumed by its authors. The work of P. Latafat was supported by the Research Foundation Flanders (FWO) grants 1196820N and 12Y7622N. The work of P. Patrinos was supported by the Research Foundation Flanders (FWO) research projects G0A0920N, G086518N, G086318N, and G081222N; Research Council KU Leuven C1 project No. C14/18/068; Fonds de la Recherche Scientifique - FNRS and the Fonds Wetenschappelijk Onderzoek - Vlaanderen under EOS project 30468160 (SeLMA); European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 953348. The work of A. Themelis was supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI grant JP21K17710.

[†]KU Leuven, Department of Electrical Engineering (ESAT-STADIUS), Kasteelpark Arenberg 10, 3001 Leuven, Belgium. Emails: {pourya.behmandpoor,puya.latafat,marc.moonen,panos.patrinos}@esat.kuleuven.be

[‡]Kyushu University, Faculty of Information Science and Electrical Engineering (ISEE), 744 Motoooka Nishi-ku, 819-0395 Fukuoka, Japan. Email: andreas.themelis@ees.kyushu-u.ac.jp

The minimization problem (1.1) has gained considerable attention across various disciplines including machine learning (ML), signal and image processing, statistics, and control. Despite an upsurge in developing optimization methods to address such a problem, the potential of low-memory quasi-Newton methods has largely been neglected which can be partially attributed to the absence of theoretical foundations for handling nonsmooth settings. In the smooth strongly convex settings, competitive convergence rates compared to typical ML approaches have been documented in the ML community [47]. This work aims to address such large-scale problems in their full generality in the nonconvex, nonsmooth problem settings.

Stochastic gradient descent (SGD) is commonly employed for finite sum minimization problems. Despite it involving simple iterations, SGD requires a diminishing stepsize and, even in the strongly convex setting, can only achieve sublinear rates of convergence. These limitations have prompted the development of several stochastic and incremental methods such as SAG [61], SAGA [21], SDCA [62], SPIDER [28], SVRG [36] and its extensions [55, 29], SARAH [51], and zeroSARAH [41], which primarily target smooth functions ($g = 0$) and are often restricted to the convex regime. To accommodate composite nonsmooth cost functions of the form (1.1), studies such as [14], proxSGD [31], proxSAGA and proxSVRG [56], proxSARAH [53], and SpiderBoost [70] have emerged recently.

The majority of the methods mentioned above incorporate a combination of stochastic and deterministic components in addressing the finite sum problem, aiming to diminish the variance of iterates toward the optimal point. Notably, algorithms such as SAGA, SVRG, and SARAH employ an outer loop to incorporate full gradients as the deterministic enhancement, along with an inner loop that incorporates stochastic gradients using randomized sampling with replacement. Furthermore, these algorithms adopt fixed stepsizes, in contrast to SGD which necessitates diminishing stepsizes to mitigate variance. In line with the spirit of these methods, the proposed algorithm also utilizes both inner and outer loops.

In its inner loop, the algorithm investigated in this study can be perceived as an incremental approach with a (shuffled) cyclic (randomized without replacement) sweeping rule. It should be mentioned that when combined with SGD, this sweeping rule demonstrates superior convergence and implementation efficiency [54, 5] compared to the randomized sweeping rule with replacement. Moreover, the analysis of SGD with (randomized) sampling without replacement has recently emerged in convex regimes, providing enhanced bounds compared to standard SGD [13, 16, 44, 32, 34]. Beyond SGD, the proposed algorithm, in its basic form without a linesearch, can be considered as a memory-efficient variant of Finito [22] and MISO [43]. It is worth noting that DIAG [46], proposed independently, studies the Finito/MISO algorithm under a cyclic sweeping rule and in the strongly convex case. More recently, [40] provided a comprehensive study of the above algorithms in the fully nonconvex setting. However, the above are all limited to first-order methods.

In its outer loop, one distinguishing characteristic of the proposed algorithm, which sets it apart from stochastic algorithms like SVRG and SARAH, is its utilization of quasi-Newton directions integrated with a linesearch while preserving the advantageous low-memory characteristic. In the context of this study, various methodologies have been explored to address nonconvex nonsmooth composite functions by employing quasi-Newton directions. For instance, methods presented in [68, 66, 1] have demonstrated the application of quasi-Newton directions to achieve superlinear convergence rates, albeit limited to scenarios involving a single smooth function within the composite cost. In the finite sum setting, approaches proposed by [47] and [60] have utilized quasi-Newton updates with global convergence guarantees and linear convergence rates. Furthermore, [74] has extended the utilization of quasi-Newton directions to decentralized learning scenarios.

To attain a superlinear convergence rate, the IQN method [45] has integrated quasi-Newton directions with incremental updates, albeit with only local convergence guarantees. Conversely, the approach introduced in [59] also exhibits a superlinear convergence rate but necessitates Hessian evaluation. It is noteworthy that the aforementioned algorithms are applicable in (strongly) convex cases. However, within the nonconvex nonsmooth setting, the algorithm proposed by [71] stands out with global convergence guarantees when the nonsmooth term is convex.

One of the restrictive aspects of the aforementioned works is that the cost functions are Lipschitz differ-

entiable. However, in numerous practical applications, although the cost functions are differentiable, they fail to satisfy Lipschitz continuity assumptions on their gradients. This issue is exemplified in [Section 5.2](#). To tackle such cost functions, the proposed algorithm goes beyond the classical notion of smoothness and employs the concept of relative smoothness, as introduced in [\[3, 42\]](#). In connection with the proposed method, stochastic mirror descent (SMD) methods incorporate relative smoothness. Notable references in this area include [\[4, 48, 33, 19\]](#). Within the convex setting, PLIAG [\[73\]](#) has been introduced as the Bregman variant of IAG [\[7, 8, 69\]](#). Furthermore, [\[25\]](#) explores Bregman stochastic gradient descent (BSGD). In the nonconvex regime, [\[39\]](#) investigates a Bregman variant of Finito/MISO.

While the literature often assumes convexity for the nonsmooth term g , our proposed method, as indicated in [Assumption 1](#), allows the nonconvex nature of this term. This enables the algorithm to effectively handle a wide range of nonconvex constraints, including rank constraints and ℓ_0 -norm ball constraints, as well as nonconvex regularizers such as ℓ^p with $p \in [0, 1)$.

Motivated by the aforementioned advancements and recognizing the existing limitations in the literature, the proposed method addresses the optimization of regularized nonsmooth nonconvex cost functions, allowing the gradients of differentiable functions in the finite sum to be non-Lipschitz. To the best of our knowledge, none of the currently available methods in the literature that exhibit superlinear convergence rates have explicitly addressed the challenge of handling non-Lipschitz differentiable functions within nonsmooth nonconvex finite sum settings.

Contributions

The main contributions of the paper are as follows:

1. We propose SPIRAL with convergence guarantees for a wide class of finite sum problems. Not only are both the nonsmooth regularizer g and the finite sum terms f_i all allowed to be nonconvex, but also f_i functions do not need to have Lipschitz-continuous gradients. Moreover, unlike Finito/MISO/DIAG, SPIRAL requires only $\mathcal{O}(n)$ memory allocation.
2. When the nonsmooth term is convex, we show that SPIRAL converges superlinearly when the employed quasi-Newton directions are superlinear (cf. [Definition 4.11](#)) and the linesearch will eventually never be invoked (cf. [Theorem 4.12](#)) under mild assumptions. This is also supported by our simulation results. Moreover, global (as opposed to local) convergence is guaranteed regardless of any assumptions placed on the quasi-Newton directions or the convexity of the nonsmooth term (cf. [Theorem 4.7](#)).
3. Finally, an adaptive variant employing appropriate backtracking linesearch is introduced that adapts to the local relative smoothness moduli of f_i while maintaining convergence guarantees and the $\mathcal{O}(n)$ memory requirement.

2 Preliminaries

2.1 Notation

In this section, we provide basic notations. The interested reader may refer to [\[58, 57\]](#) for details. The set of natural numbers is denoted by $\mathbb{N} = \{0, 1, 2, \dots\}$. The set of real and extended-real numbers are $\mathbb{R} := (-\infty, \infty)$ and $\overline{\mathbb{R}} := \mathbb{R} \cup \{\infty\}$, and the set of positive reals is denoted by $\mathbb{R}_+ := [0, \infty)$. We also use the notation $[N] := \{1, 2, \dots, N\}$. We denote by $\langle \cdot, \cdot \rangle$ and $\|\cdot\|$ the standard Euclidean inner product and the induced norm. The distance of a point $x \in \mathbb{R}^n$ to a nonempty set $S \subseteq \mathbb{R}^n$ is given by $\text{dist}(x, S) = \inf_{z \in S} \|z - x\|$. For a vector $\mathbf{w} = (w_1, \dots, w_r) \in \mathbb{R}^{\sum_i n_i}$, $w_i \in \mathbb{R}^{n_i}$ is used to denote its i -th block coordinate. The identity operator is denoted by id .

For a sequence $(x^k)_{k \in \mathbb{N}}$ we write $(x^k)_{k \in \mathbb{N}} \subseteq E$ to indicate that $x^k \in E$ for all $k \in \mathbb{N}$. We use the following notions of convergence rate: a sequence $(x^k)_{k \in \mathbb{N}}$ is said to converge to a point x^* :

- (at least) Q -linearly (with quotient rate) with Q -factor given by $\sigma \in (0, 1)$, if there exists $k_0 \in \mathbb{N}$ such that for all $k \geq k_0$,

$$\|x^{k+1} - x^*\| \leq \sigma \|x^k - x^*\|.$$

- (at least) R -linearly (with root rate) if there exists a sequence of nonnegative scalars $(v^k)_{k \in \mathbb{N}}$ such that $\|x^k - x^*\| \leq v^k$ and $(v^k)_{k \in \mathbb{N}}$ converges Q -linearly to zero.
- superlinearly if either $x^k = x^*$ for some $k \in \mathbb{N}$ or

$$\lim_{k \rightarrow \infty} \frac{\|x^{k+1} - x^*\|}{\|x^k - x^*\|} = 0.$$

We use the notation $Q : \mathbb{R}^n \rightrightarrows \mathbb{R}^m$ to indicate a mapping from each point $x \in \mathbb{R}^n$ to a subset $Q(x)$ of \mathbb{R}^m . The *graph* of Q is the set $\text{gph } Q := \{(x, y) \in \mathbb{R}^n \times \mathbb{R}^m : y \in Q(x)\}$, and the set of its *fixed points* is defined as $\text{fix } Q := \{x \in \mathbb{R}^n : x \in Q(x)\}$. We say that Q is *outer semicontinuous* (*osc*) if $\text{gph } Q$ is a closed subset of $\mathbb{R}^n \times \mathbb{R}^m$, and *locally bounded* if for every bounded $U \subset \mathbb{R}^n$ the set $\bigcup_{x \in U} Q(x)$ is bounded.

The *domain* of an extended-real-valued function $\phi : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$ is the set $\text{dom } \phi := \{x \in \mathbb{R}^n : \phi(x) < \infty\}$ and $\text{epi } \phi := \{(x, \alpha) \in \mathbb{R}^n \times \mathbb{R} : \phi(x) \leq \alpha\}$ is its *epigraph* set. Function ϕ is said to be *proper* if $\text{dom } \phi \neq \emptyset$, and *lower semicontinuous* (*lsc*) if $\text{epi } \phi$ is a closed subset of \mathbb{R}^{n+1} . We say that ϕ is *level bounded* if its α -sublevel set $\text{lev}_{\leq \alpha} \phi := \{x \in \mathbb{R}^n : \phi(x) \leq \alpha\}$ is bounded for all $\alpha \in \mathbb{R}$. The indicator function δ_X of a nonempty set $X \subseteq \mathbb{R}^n$ is defined by

$$\delta_X(x) := \begin{cases} 0 & \text{if } x \in X \\ +\infty & \text{if } x \notin X. \end{cases} \quad (2.1)$$

The indicator function δ_X is closed if and only if X is a closed set.

We denote by $\hat{\partial}\phi : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ the *regular sub-differential* of ϕ , where

$$v \in \hat{\partial}\phi(\bar{x}) \iff \liminf_{\bar{x} \neq x \rightarrow \bar{x}} \frac{\phi(x) - \phi(\bar{x}) - \langle v, x - \bar{x} \rangle}{\|x - \bar{x}\|} \geq 0.$$

The regular sub-differential is closed- and convex-valued. The (limiting) *sub-differential* of ϕ is $\partial\phi : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$, where $v \in \partial\phi(x)$ iff $x \in \text{dom } \phi$ and there exists a sequence $(x^k, v^k)_{k \in \mathbb{N}} \subseteq \text{gph } \hat{\partial}\phi$ such that $(x^k, \phi(x^k), v^k) \rightarrow (x, \phi(x), v)$ as $k \rightarrow \infty$.

A necessary condition for local minimality of x for ϕ is $0 \in \hat{\partial}\phi(x)$, see [57, Thm. 10.1]. Finally, the set of r times continuously differentiable functions over \mathbb{R}^n is denoted by $\mathcal{C}^r = \mathcal{C}^r(\mathbb{R}^n)$.

2.2 Relative smoothness

We start by formally defining the notion of distance generating function, Bregman distance, and relative smoothness [3, 42, 65].

Definition 2.1 (Distance-generating function (dgf)). *A strictly convex function $h : \mathbb{R}^n \rightarrow \mathbb{R}$ that is continuously differentiable everywhere will be referred to as a distance-generating function (dgf).*

While some results here can be presented with $\text{dom } h \subset \mathbb{R}^n$, for the sake of simplicity and global convergence analysis, we continue with $\text{dom } h = \mathbb{R}^n$. The Bregman distance associated with a dgf is defined as:

Definition 2.2 (Bregman distance). *Given a dgf $h : \mathbb{R}^n \rightarrow \mathbb{R}$, the Bregman distance $D_h : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}_+$ is defined as,*

$$D_h(y, x) := h(y) - h(x) - \langle \nabla h(x), y - x \rangle \quad \text{for all } x, y \in \mathbb{R}^n. \quad (2.2)$$

Definition 2.3 (Relative smoothness [12, Def. 2.2]). *A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is smooth relative to a dgf $h : \mathbb{R}^n \rightarrow \mathbb{R}$ if there exists $L \geq 0$ such that $Lh \pm f$ are convex functions on \mathbb{R}^n . In this case, we say that f is L -smooth (relative to h) to make the modulus L explicit.*

Fact 2.4 (Descent lemma [12, Lem. 2.1]). *If $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is L -smooth relative to a dgf $h : \mathbb{R}^n \rightarrow \mathbb{R}$, then for all $x, y \in \mathbb{R}^n$*

$$|f(y) - f(x) - \langle \nabla f(x), y - x \rangle| \leq L D_h(y, x).$$

Note that in the Euclidean case, the dgf and the corresponding Bregman distance reduce to $h = \frac{1}{2} \|\cdot\|^2$ and $D_h(y, x) = \frac{1}{2} \|y - x\|^2$, respectively, and Fact 2.4 reduces to the ordinary descent lemma [6, Prop. A.24] for smooth functions.

Relative to a dgf $h : \mathbb{R}^n \rightarrow \mathbb{R}$, the (left) Bregman proximal mapping of a proper lsc function $\phi : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$ is the set-valued mapping $\text{prox}_\phi^h : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ defined as [37, Def. 2.2]

$$\text{prox}_\phi^h(x) := \text{argmin}_{w \in \mathbb{R}^n} \{\phi(w) + D_h(w, x)\}, \quad (2.3)$$

and its value function is the Bregman Moreau envelope $\phi^h : \mathbb{R}^n \rightarrow \mathbb{R}$

$$\phi^h(x) := \min_{w \in \mathbb{R}^n} \{\phi(w) + D_h(w, x)\}. \quad (2.4)$$

Moreover it is evident from (2.3) and Definition 2.2 that if $v \in \text{prox}_\phi^h(x)$, then

$$\nabla h(x) - \nabla h(v) \in \hat{\partial} \phi(v), \quad (2.5)$$

and the converse also holds when ϕ is convex. Whenever the superscript h is omitted from prox_ϕ^h , it refers to the Euclidean proximal mapping with $h = \frac{1}{2} \|\cdot\|^2$.

3 Proposed algorithm

The proposed method, SPIRAL, is outlined in Algorithm 1 to address the optimization problem (1.1). SPIRAL employs the set-valued mapping $t : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ as its major oracle, defined as

$$t(s) := \text{argmin}_{w \in \mathbb{R}^n} \left\{ g(w) + \sum_{i=1}^N \frac{1}{\gamma_i} h_i(w) - \langle s, w \rangle \right\}, \quad (3.1)$$

where h_i is the dgf corresponding to function f_i as in Assumption 1.A1. Within both the outer and inner loops, the mapping t , utilized in steps 1.1, 1.3, 1.5.c, and 1.9, represents the proximal steps. It is important to note that in the case of Euclidean space, where the functions f_i are L_i -smooth relative to $h_i = \frac{1}{2} \|\cdot\|^2$, the oracle t reduces to the Euclidean proximal mapping with updates of the form $z^k \in \text{prox}_{\hat{\gamma}g}(\hat{\gamma}s^k)$, where $\hat{\gamma}^{-1} = \sum_{i=1}^N \gamma_i^{-1}$. Similarly, the iterates v^k , y^k , and \tilde{z}^k are updated using the same $\text{prox}_{\hat{\gamma}g}$ function. A detailed description of the Euclidean version of the algorithm can be found in Appendix E.2.

SPIRAL employs the function $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ defined by

$$\mathcal{L}(y, x) := \varphi(y) + D_{\hat{h}}(y, x) \quad \text{with } y \in t(\nabla \hat{h}(x)) \quad (3.2)$$

in its linesearch, where $\hat{h}_i : \mathbb{R}^n \rightarrow \mathbb{R}$ and $\hat{h} : \mathbb{R}^n \rightarrow \mathbb{R}$ are

$$\hat{h}_i := \frac{1}{\gamma_i} h_i - \frac{1}{N} f_i, \quad \hat{h} := \sum_{i=1}^N \hat{h}_i. \quad (3.3)$$

The function \mathcal{L} is considered as a suitable Lyapunov function in our convergence analysis. The linesearch in step 1.5 interpolates the iterate u^k between the candidate fast update $z^k + d^k$, corresponding to $\tau_k = 1$, and the safeguard step v^k which is approached as $\tau_k \searrow 0$. The iterate u^k is selected whenever it is a descent direction for the Lyapunov function \mathcal{L} (cf. Remark 4.5 regarding the well definedness of the linesearch).

Algorithm 1 SPIRAL

Initialize $z^{\text{init}} \in \mathbb{R}^n$, $\gamma_i \in (0, N/L_i)$, $\forall i \in [N]$, $s^0 = \sum_{i=1}^N \frac{1}{\gamma_i} \nabla h_i(z^{\text{init}}) - \frac{1}{N} \nabla f_i(z^{\text{init}})$,
 maximum number of backtracks $q_{\max} \in \mathbb{N} \cup \{\infty\}$ (e.g. $q_{\max} = 2$), $\beta \in (0, 1)$

Repeat for $k = 0, 1, \dots, K$

1.1: $z^k \in \mathfrak{t}(s^k)$

1.2: $\bar{s}^k = \sum_{i=1}^N \frac{1}{\gamma_i} \nabla h_i(z^k) - \frac{1}{N} \nabla f_i(z^k)$ (full update)

1.3: $v^k \in \mathfrak{t}(\bar{s}^k)$

1.4: choose $d^k \in \mathbb{R}^n$ at z^k (e.g. based on a quasi-Newton method for solving $r_{\hat{h}}(z) = 0$)

1.5: set $\tau_k = 1, q_k = 0$ (linesearch)

a: $u^k = \tau_k z^k + (1 - \tau_k)v^k + \tau_k d^k$

b: $\tilde{s}^k = \sum_{i=1}^N \frac{1}{\gamma_i} \nabla h_i(u^k) - \frac{1}{N} \nabla f_i(u^k)$ (full update)

c: $y^k \in \mathfrak{t}(\tilde{s}^k)$

d: **if** $\mathcal{L}(y^k, u^k) \leq \mathcal{L}(v^k, z^k)$
 go to step 1.6

e: **else if** $q_k = q_{\max}$ **then**
 $u^k = v^k$, $\tilde{s}^k = \sum_{i=1}^N \frac{1}{\gamma_i} \nabla h_i(u^k) - \frac{1}{N} \nabla f_i(u^k)$, and go to step 1.6

f: **else**
 $\tau_k \leftarrow \beta \tau_k$, $q_k \leftarrow q_k + 1$, and go to step 1.5.a

1.6: $s^k \leftarrow \tilde{s}^k$

1.7: **for** $\ell = 1, \dots, N$ **do** (incremental loop)

1.8: randomly choose $i^\ell \in [N]$ without replacement

1.9: $\tilde{z}_{i^\ell}^k \in \mathfrak{t}(s^k)$

1.10: $s^k \leftarrow s^k + \left[\frac{1}{\gamma_{i^\ell}} \nabla h_{i^\ell}(\tilde{z}_{i^\ell}^k) - \frac{1}{N} \nabla f_{i^\ell}(\tilde{z}_{i^\ell}^k) - \frac{1}{\gamma_{i^\ell}} \nabla h_{i^\ell}(u^k) + \frac{1}{N} \nabla f_{i^\ell}(u^k) \right]$

1.11: $s^{k+1} \leftarrow s^k$

Return z^K

One distinguishing characteristic of SPIRAL, which sets it apart from stochastic algorithms such as SVRG and SARAH, is its utilization of directions d^k in step 1.4 based on second-order-like information of the set-valued residual mapping $r_{\hat{h}} : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ defined as

$$r_{\hat{h}} := \text{id} - \mathfrak{t} \circ \nabla \hat{h}. \quad (3.4)$$

This feature allows SPIRAL to achieve a superlinear convergence rate, given certain mild conditions (cf. [Theorem 4.12](#)). Given that the inclusion $0 \in r_{\hat{h}}(z^*)$ holds at a stationary point z^* (as discussed in [Section 4.1](#) and expressed in (4.7)), the direction d^k is determined based on solving this inclusion. Semismooth Newton directions [66] can be utilized to compute d^k ; however, this approach relies on access to second-order oracle information. As an alternative to circumvent this requirement, quasi-Newton methods can be employed to compute the directions d^k . Specifically, we employ the update rule

$$d^k = -H^k r_{\hat{h}}(z^k) \quad (3.5)$$

where H^k represents a linear operator that approximates the second-order information of the residual mapping $r_{\hat{h}}$. It is worth noting that although $r_{\hat{h}}$ is a set-valued mapping, it typically exhibits single-valuedness and other desirable properties in the vicinity of stationary points of the objective function φ (see e.g. [1, Thm. 3.10 and 3.11]). Consequently, the updates for H^k can be implemented using popular quasi-Newton methods such as Broyden's method, BFGS, and L-BFGS (cf. [Section 4.4](#) for further discussion).

SPIRAL can accommodate various sweeping rules depending on the memory requirements. The following remark comments on two different settings.

Remark 3.1 (sweeping rule in the incremental loop).

- (i) *low-memory setting*: in this setting, SPIRAL employs a (shuffled) cyclic (randomized without replacement) sweeping rule within the incremental loop, and unlike Finito/MISO/DIAG methods, it does not require storing individual function gradients ∇f_i . Instead, SPIRAL only requires storing the finite sum of gradients as $\nabla \hat{h}$, using the vectors s^k , \bar{s}^k , and $\bar{\bar{s}}^k$. Additionally, in the incremental loop, updating this sum in the vector s^k is carried out in a memory-efficient manner since the vector u^k is known and fixed due to choosing i^ℓ without replacement. This advantageous characteristic allows SPIRAL to require only $\mathcal{O}(n)$ memory allocation, making it suitable for large-scale optimization problems.
- (ii) *high-memory setting*: in this setting, by utilizing a memory of $\mathcal{O}(nN)$ and saving \tilde{z}_j^k for all $j \in [N]$, like Finito/MISO/DIAG, the incremental loop can be replaced by a randomized loop of an arbitrary depth.

Computational complexity. The overall computational complexity, measured in terms of gradient evaluations per iteration, is $\mathcal{O}((4 + \kappa)N)$. This includes $2 + \kappa$ full gradient evaluations performed outside of the inner loop, where κ represents the number of backtracks in the linesearch, and two gradient evaluations performed in each iteration of the inner loop. It is worth noting that in certain problems like least squares, nonnegative principal component analysis, and logistic regression, the gradients can be obtained by storing the inner product between data points and the evaluated points. Consequently, the gradient evaluations $\nabla \hat{h}_i(u^k)$ in step 1.10 can be derived from the computations in step 1.5.b, thereby reducing the computational complexity to $\mathcal{O}((3 + \kappa)N)$. In addition, based on numerical experiments, it is beneficial to limit the number of backtracks using a maximum value q_{\max} . It is important to note that, as demonstrated in Theorem 4.12, under mild conditions, the unit stepsize $\tau_k = 1$ is eventually always accepted ($\kappa = 0$), allowing for pure quasi-Newton type updates in step 1.5.a and avoiding any further backtracks and computation.

4 Convergence Analysis

In the subsequent subsections, we study the convergence of SPIRAL by reformulating the problem (1.1) in the lifted space. We will thus recast the Lyapunov function \mathcal{L} in (3.2) into this new space and study it along the iterates generated by SPIRAL. In the lifted space, we can show that block coordinate updates in the incremental loop result in sufficient descent for the Lyapunov function, while this is not necessarily the case for the objective function. Utilizing newly reformulated operators, we then outline Algorithm 1 in the lifted space. Having new insight into the mechanism of SPIRAL, in Sections 4.3 to 4.5 we establish its convergence in various regimes.

4.1 Problem Reformulation

We recast (1.1) as the following lifted consensus optimization problem:

$$\text{minimize}_{\mathbf{z}=(z_1, \dots, z_N) \in \mathbb{R}^{Nn}} \Phi(\mathbf{z}) := \underbrace{\frac{1}{N} \sum_{i=1}^N f_i(z_i)}_{:=F(\mathbf{z})} + \underbrace{\frac{1}{N} \sum_{i=1}^N g(z_i)}_{:=G(\mathbf{z})} + \delta_\Delta(\mathbf{z}) \quad (4.1)$$

$$\text{where } \Delta := \{ \mathbf{z} = (z_1, \dots, z_N) \in \mathbb{R}^{Nn} \mid z_1 = \dots = z_N \}$$

is the *consensus set*. Note that $\Phi(\mathbf{z}) = \varphi(\mathbf{z})$ whenever $\mathbf{z} = (z, \dots, z) \in \Delta$. Define

$$\mathcal{M}_{\hat{H}}(\mathbf{w}, \mathbf{z}) := \Phi(\mathbf{w}) + D_{\hat{H}}(\mathbf{w}, \mathbf{z}), \quad \text{with } \hat{H}(\mathbf{x}) := \sum_{i=1}^N \hat{h}_i(x_i), \quad (4.2)$$

for any $(\mathbf{w}, \mathbf{z}) \in \mathbb{R}^{Nn} \times \mathbb{R}^{Nn}$, where \hat{h}_i is defined in (3.3). The model $\mathcal{M}_{\hat{H}}$ is in particular a *majorizing model* of Φ , in that from (3.3), and Definitions 2.1 and 2.3, it is evident that whenever $\gamma_i \in (0, N/L_i)$, \hat{h}_i and consequently \hat{H} are dgfs, hence $D_{\hat{H}}(\mathbf{w}, \mathbf{z}) \geq 0$ for all $\mathbf{w}, \mathbf{z} \in \mathbb{R}^{Nn}$ and

- (i) $\mathcal{M}_{\hat{H}}(\mathbf{w}, \mathbf{z}) \geq \Phi(\mathbf{w})$ for all $\mathbf{w}, \mathbf{z} \in \mathbb{R}^{Nn}$;
- (ii) $\mathcal{M}_{\hat{H}}(\mathbf{z}, \mathbf{z}) = \Phi(\mathbf{z})$ for all $\mathbf{z} \in \mathbb{R}^{Nn}$.

The *Bregman proximal mapping* $\text{prox}_{\Phi}^{\hat{H}} : \mathbb{R}^{Nn} \rightrightarrows \mathbb{R}^{Nn}$ and its value function the *Bregman Moreau envelope* $\Phi^{\hat{H}} : \mathbb{R}^{Nn} \rightarrow \mathbb{R}$ (recall the definitions in (2.3)) are then defined as

$$\text{prox}_{\Phi}^{\hat{H}}(\mathbf{z}) := \underset{\mathbf{w} \in \mathbb{R}^{Nn}}{\text{argmin}} \mathcal{M}_{\hat{H}}(\mathbf{w}, \mathbf{z}), \quad \Phi^{\hat{H}}(\mathbf{z}) := \min_{\mathbf{w} \in \mathbb{R}^{Nn}} \mathcal{M}_{\hat{H}}(\mathbf{w}, \mathbf{z}). \quad (4.3)$$

The corresponding *forward-backward residual* is defined as $R_{\hat{H}} = \text{id} - \text{prox}_{\Phi}^{\hat{H}}$. The envelope $\Phi^{\hat{H}}$ is considered as the Lyapunov function in our convergence studies.

We proceed with the following fact that is the key to our convergence analysis. First, to facilitate our subsequent analysis, we introduce the matrix $U_i \in \mathbb{R}^{Nn \times n}$ whose i -th block rows form an identity matrix. For a given vector $\mathbf{y} \in \mathbb{R}^n$, the action of U_i can be expressed as:

$$U_i \mathbf{y} = (0, \dots, 0, \overset{i\text{-th block}}{\overline{\mathbf{y}}}, 0, \dots, 0). \quad (4.4)$$

The following fact demonstrates that block-coordinate updates result in descent on the Bregman Moreau envelope $\Phi^{\hat{H}}$ in (4.3). As remarked above this is not necessarily the case for the cost function.

Fact 4.1 (Descent lemma [39, Lem. 4.2]). *Suppose that Assumption 1 holds, and let $\mathbf{x} = (x_1, \dots, x_N)$. Fix $\mathbf{y} \in \text{prox}_{\Phi}^{\hat{H}}(\mathbf{x})$, and let $\mathcal{I} \subseteq [N]$ be a subset of indices. Consider the block-coordinate update*

$$\mathbf{v} = \mathbf{x} + \sum_{i \in \mathcal{I}} U_i U_i^{\top} (\mathbf{y} - \mathbf{x}), \quad (4.5)$$

where U_i is as defined in (4.4). Then,

$$\Phi^{\hat{H}}(\mathbf{v}) \leq \Phi^{\hat{H}}(\mathbf{x}) - D_{\hat{H}}(\mathbf{v}, \mathbf{x}). \quad (4.6)$$

This observation is then utilized to establish that the limit points of the sequence $(z^k)_{k \in \mathbb{N}}$ correspond to stationary points of the function φ , which, in the nonconvex setting, represents the necessary condition $0 \in \hat{\partial}\varphi(z^*)$.

In the following fact we present some of useful properties of the Bregman proximal mapping $\text{prox}_{\Phi}^{\hat{H}}$ and the Bregman distance $D_{\hat{H}}$, and expand on their relation to those defined in Section 3.

Fact 4.2 (Bregman proximal mapping [39, Lem. 3.1]). *Suppose that Assumption 1 holds and let $\gamma_i \in (0, \frac{N}{L_i})$. Then, the following hold:*

- (i) $D_{\hat{H}}(\mathbf{v}, \mathbf{z}) \geq \sum_{i=1}^N (\frac{1}{\gamma_i} - \frac{L_{f_i}}{N}) D_{h_i}(v_i, z_i)$, for $(\mathbf{v}, \mathbf{z}) \in \mathbb{R}^{Nn} \times \mathbb{R}^{Nn}$.
- (ii) $\text{prox}_{\Phi}^{\hat{H}}(\mathbf{z}) = \{(v, \dots, v) : v \in \text{t}(\sum_{i=1}^N \nabla \hat{h}_i(z_i))\}$, where t is as in (3.1), is a nonempty and compact subset of Δ .

When $\mathbf{z} \in \Delta$, one has a lower-dimensional representation of the Bregman Moreau envelope $\Phi^{\hat{H}}$, provided in the following corollary of [Fact 4.2](#).

Corollary 4.3 (lower-dimensional representations). *Let [Assumption 1](#) hold and let $\gamma_i \in (0, \frac{N}{L_i})$. Then, with the Bregman Moreau operator and the envelope associated with φ in [\(1.1\)](#) given by*

$$\begin{aligned} \mathfrak{t}_{\hat{h}}(z) &= \operatorname{argmin}_{w \in \mathbb{R}^n} \{ \varphi(w) + D_{\hat{h}}(w, z) \}, \\ \varphi^{\hat{h}}(z) &= \min_{w \in \mathbb{R}^n} \{ \varphi(w) + D_{\hat{h}}(w, z) \} = \mathcal{L}(v, z), \text{ with } v \in \mathfrak{t}_{\hat{h}}(z), \end{aligned}$$

it holds that $\mathfrak{t}_{\hat{h}} = \mathfrak{t} \circ \nabla \hat{h} = \operatorname{prox}_{\varphi}^{\hat{h}}$, and $\Phi^{\hat{H}}(\mathbf{z}) = \varphi^{\hat{h}}(z)$ for $\mathbf{z} = (z, \dots, z) \in \Delta$. Moreover,

(i) $D_{\hat{h}}(v, z) \geq \sum_{i=1}^N (\frac{1}{\gamma_i} - \frac{L_{f_i}}{N}) D_{h_i}(v, z)$, for $(v, z) \in \mathbb{R}^n \times \mathbb{R}^n$.

(ii) If $v \in \operatorname{prox}_{\varphi}^{\hat{h}}(z)$, then $\nabla \hat{h}(z) - \nabla \hat{h}(v) \in \hat{\partial} \varphi(v)$; the converse also holds when φ is convex.

An important consequence of [Fact 4.2](#) and its [Corollary 4.3](#) is that the range of $\operatorname{prox}_{\Phi}^{\hat{H}}$ is a subset of the consensus set Δ (cf. [Fact 4.2\(ii\)](#)). Moreover, by [Corollary 4.3\(ii\)](#), to any fixed point of $\operatorname{prox}_{\Phi}^{\hat{H}}$ (or $\operatorname{prox}_{\varphi}^{\hat{h}}$) there corresponds a stationary point for the original problem. That is to say

$$\begin{aligned} z^* \in \operatorname{fix} \operatorname{prox}_{\varphi}^{\hat{h}} &\Leftrightarrow 0 \in \mathfrak{r}_{\hat{h}}(z^*) \Leftrightarrow z^* \in \operatorname{prox}_{\varphi}^{\hat{h}}(z^*) \\ &\Leftrightarrow (z^*, \dots, z^*) = \mathbf{z}^* \in \operatorname{prox}_{\Phi}^{\hat{H}}(\mathbf{z}^*) \\ &\Leftrightarrow 0 \in \mathfrak{R}_{\hat{H}}(\mathbf{z}^*) \Leftrightarrow 0 \in \hat{\partial} \Phi(\mathbf{z}^*) \\ &\Leftrightarrow 0 \in \hat{\partial} \varphi(z^*), \end{aligned} \tag{4.7}$$

where $\mathfrak{r}_{\hat{h}}$ is defined in [\(3.4\)](#).

4.2 Lifted Representation of the Algorithm

For the sake of clarity in presentation and without loss of generality, we consider the cyclic sweeping rule in the incremental loop where $i^\ell = \ell$ in step [1.8](#) (cf. [Remark 4.6](#) for shuffled cyclic sweeping rule). In this case, we adopt the following notation:

$$\bar{\mathbf{z}}_\ell^k := (\bar{z}_1^k, \bar{z}_2^k, \dots, \bar{z}_{\ell-1}^k, \underbrace{u^k, \dots, u^k}_{N-\ell+1}), \quad \ell \in [N]. \tag{4.8}$$

Using the defined operators in the previous subsection, the proposed [Algorithm 1](#) is outlined in the lifted space in [Algorithm 2](#).

In [Algorithm 2](#), $\mathbf{z}^k, \mathbf{v}^k$ belong to the consensus set Δ owing to [Fact 4.2\(ii\)](#). This along with the choice of \mathbf{d}^k implies the same for \mathbf{u}^k , ensuring that the linesearch can be performed in the lower dimensional space (see [Remark 4.5](#)). In the following proposition, we highlight the equivalence of [Algorithm 1](#) and its lifted variant [Algorithm 2](#). The proof is omitted as it follows directly from the above observations along with [Fact 4.2](#) and [Corollary 4.3](#).

Proposition 4.4. *As long as the two algorithms are initialized with the same parameters, to any sequence $(z^k, v^k, d^k, u^k, y^k, \bar{z}_\ell^k, \ell \in [N])_{k \in \mathbb{N}}$ generated by [Algorithm 1](#), there correspond sequences $(\mathbf{z}^k = (z^k, \dots, z^k))_{k \in \mathbb{N}}$, $(\mathbf{v}^k = (v^k, \dots, v^k))_{k \in \mathbb{N}}$, $(\mathbf{d}^k = (d^k, \dots, d^k))_{k \in \mathbb{N}}$, $(\mathbf{u}^k = (u^k, \dots, u^k))_{k \in \mathbb{N}}$, $(\mathbf{y}^k = (y^k, \dots, y^k))_{k \in \mathbb{N}}$, $(\bar{\mathbf{z}}_\ell^k = (\bar{z}_\ell^k, \dots, \bar{z}_\ell^k))_{k \in \mathbb{N}}$, $\ell \in [N]$, generated by [Algorithm 2](#) (and vice versa).*

Considering the updates at steps [2.1](#) and [2.8](#), the descent property established in [Fact 4.1](#) already hints as to why $\Phi^{\hat{H}}$ is employed in the backtracking linesearch procedure. In the next remark, we expand on the well-definedness of this linesearch and discuss its relation to the one prescribed in [Algorithm 1](#). These observations in the lifted space will help us in establishing convergence of the algorithm in [Sections 4.3](#) to [4.5](#).

Algorithm 2 Representation of [Algorithm 1](#) in the lifted space

Initialize $\bar{z}_N^{-1} \in \mathbb{R}^{Nn}$, $\beta \in (0, 1)$, $\gamma_i \in (0, N/L_i)$, $i \in [N]$
maximum number of backtracks $q_{\max} \in \mathbb{N} \cup \{\infty\}$ (e.g. $q_{\max} = 2$), $K \in \mathbb{N}$

Repeat for $k = 0, 1, \dots, K$

2.1: $\mathbf{z}^k \in \text{prox}_{\hat{\Phi}}^{\hat{H}}(\bar{\mathbf{z}}_N^{k-1})$

2.2: $\mathbf{v}^k \in \text{prox}_{\hat{\Phi}}^{\hat{H}}(\mathbf{z}^k)$ (full update)

2.3: choose $\mathbf{d}^k \in \Delta$

2.4: set $\tau_k = 1, q_k = 0$ (linesearch)

a: $\mathbf{u}^k = \tau_k \mathbf{z}^k + (1 - \tau_k) \mathbf{v}^k + \tau_k \mathbf{d}^k$

b: $\mathbf{y}^k \in \text{prox}_{\hat{\Phi}}^{\hat{H}}(\mathbf{u}^k)$ (full update)

c: **if** $\Phi^{\hat{H}}(\mathbf{u}^k) \leq \Phi^{\hat{H}}(\mathbf{z}^k)$
go to step 2.5

d: **else if** $q_k = q_{\max}$ **then**
 $\mathbf{u}^k = \mathbf{v}^k$, and go to step 2.5

e: **else**
 $\tau_k \leftarrow \beta \tau_k$, $q_k \leftarrow q_k + 1$, and go to step 2.4.a

2.5: $\bar{\mathbf{z}}_1^k = \mathbf{u}^k$

2.6: **for** $\ell = 1, \dots, N$ **do** (incremental loop)

2.7: $\tilde{\mathbf{z}}_\ell^k := (\tilde{z}_\ell^k, \dots, \tilde{z}_\ell^k) \in \text{prox}_{\hat{\Phi}}^{\hat{H}}(\bar{\mathbf{z}}_\ell^k)$

2.8: $\bar{\mathbf{z}}_{\ell+1}^k = \bar{\mathbf{z}}_\ell^k + U_\ell U_\ell^\top (\tilde{\mathbf{z}}_\ell^k - \bar{\mathbf{z}}_\ell^k)$

Remark 4.5 (Well definedness of linesearch). The linesearch in step 2.4 is well defined, as it always terminates in a finite number of backtracks. In this process, the iterate \mathbf{u}^k is interpolated between the candidate fast update $\mathbf{z}^k + \mathbf{d}^k$, corresponding to $\tau_k = 1$, and the safeguard step \mathbf{v}^k which is approached as $\tau_k \searrow 0$. Observe that $\mathcal{L}(\mathbf{v}^k, \mathbf{z}^k) = \varphi^{\hat{h}}(\mathbf{z}^k) = \Phi^{\hat{H}}(\mathbf{z}^k)$, as it follows from [Corollary 4.3](#), and that similarly $\mathcal{L}(\mathbf{y}^k, \mathbf{u}^k) = \varphi^{\hat{h}}(\mathbf{u}^k) = \Phi^{\hat{H}}(\mathbf{u}^k)$. Due to [Fact 4.1](#) and the continuity of the function $\Phi^{\hat{H}}$, as long as $\mathbf{z}^k \neq \mathbf{v}^k$, the inequality $\Phi^{\hat{H}}(\mathbf{v}^k) < \Phi^{\hat{H}}(\mathbf{z}^k)$ holds, hence the inequality is satisfied for τ_k small enough. In practice, it is beneficial to limit the number of backtracks using a maximum value q_{\max} , especially at initial iterations. Regardless, as it will be shown in [Theorem 4.12](#), under mild assumptions at the limit point eventually the iterate enters a region where backtracks will never be invoked.

The updates in the incremental loop are referred to as block-coordinate updates, since at step 2.8 only one block of $\bar{\mathbf{z}}_\ell^k$ is updated at each incremental loop iteration (equivalently as in step 1.10 of [Algorithm 1](#) due to [Fact 4.2\(ii\)](#)). Note that at each block update, due to choosing ℓ without replacement, the previous block value is known and equal to u^k , as depicted in (4.8). Consequently, the update in step 2.8 (equivalently in step 1.10 of [Algorithm 1](#)) can be accomplished by replacing u^k with \tilde{z}_ℓ^k , thereby requiring a memory of $\mathcal{O}(n)$ instead of $\mathcal{O}(nN)$. However, opting for a memory of $\mathcal{O}(nN)$ and saving \tilde{z}_j^k for all $j \in [N]$ allows the algorithm to adopt the randomized sweeping rule with replacement as well (cf. [Remark 3.1](#)). Additionally, the block updates in step 2.7 are computationally inexpensive, since if implemented by steps 1.9 and 1.10 of [Algorithm 1](#) (using [Fact 4.2\(ii\)](#)), the gradient of only one function f_ℓ is computed in each incremental iteration.

Remark 4.6 (shuffled cyclic (randomized without replacement) sweeping rule). It is evident from step 2.8 that the incremental loop can be easily modified to accommodate the shuffled cyclic (randomized without replacement) sweeping rule, as was commented for [Algorithm 1](#). To achieve this, we can consider the following update in place of step 2.8:

$$\bar{\mathbf{z}}_{\ell+1}^k = \bar{\mathbf{z}}_\ell^k + U_{i^\ell} U_{i^\ell}^\top (\tilde{\mathbf{z}}_\ell^k - \bar{\mathbf{z}}_\ell^k),$$

where $i^\ell \in [N]$ is randomly chosen without replacement.

4.3 Global and Subsequential Convergence

SPIRAL is globally (as opposed to locally) convergent whenever [Assumption 1](#) holds, without any additional assumption on the convexity of nonsmooth term g . Thanks to the proposed linesearch in step 1.5, global convergence is also guaranteed with any direction d^k derived in step 1.4, although ultimately a fast convergence rate is only achieved by employing an educated direction and under assumptions at the limit point. Motivated by [Fact 4.1](#) and [Corollary 4.3](#) and consistent with previous studies such as [\[65, 39\]](#), the Bregman Moreau envelope $\Phi^{\hat{H}}$ in (4.3) is employed as the Lyapunov function which reduces to $\mathcal{L}(v^k, z^k)$ in the original space. This function has nice properties which enables us to study the global and subsequential convergence of SPIRAL, in the next theorem:

Theorem 4.7. (*Global and subsequential convergence*) *Suppose that [Assumption 1](#) holds. The following holds for the sequence $(z^k)_{k \in \mathbb{N}}$ generated by [Algorithm 1](#):*

- (i) $\mathcal{L}(v^{k+1}, z^{k+1}) \leq \mathcal{L}(v^k, z^k) - \sum_{i=1}^N D_{\hat{h}_i}(z^{k+1}, \bar{z}_i^k)$ for $k \in \mathbb{N}$, with $\hat{h}_i = \frac{1}{\gamma_i} h_i - \frac{1}{N} f_i$;
- (ii) $(D_{\hat{h}_i}(z^{k+1}, \bar{z}_i^k))_{k \in \mathbb{N}} \rightarrow 0, i \in [N]$;
- (iii) $(\mathcal{L}(v^k, z^k))_{k \in \mathbb{N}}$ and $(\varphi(z^k))_{k \in \mathbb{N}}$ converge to a value φ_* where $\varphi(z^0) \geq \varphi_* \geq \inf \varphi$;
- (iv) φ equals φ_* on all the cluster points;
- (v) all the cluster points are fixed points for $\text{prox}_{\varphi}^{\hat{h}}$, and are in particular stationary for φ ;
- (vi) if φ is level bounded, then $(z^k)_{k \in \mathbb{N}}, (\bar{z}_i^k)_{k \in \mathbb{N}}$, for $i \in [N]$ are bounded.

Proof. [4.7\(i\)](#): The block-coordinate interpretation of vectors \bar{z}_ℓ^k as shown in step 2.8 of [Algorithm 2](#) along with [Fact 4.1](#) yields

$$\Phi^{\hat{H}}(\bar{z}_{\ell+1}^k) \leq \Phi^{\hat{H}}(\bar{z}_\ell^k) - D_{\hat{H}}(\bar{z}_{\ell+1}^k, \bar{z}_\ell^k) \leq \Phi^{\hat{H}}(\bar{z}_\ell^k) \quad \text{for } \ell = 1, \dots, N-1. \quad (4.9)$$

By unrolling the inequality above we have

$$\Phi^{\hat{H}}(\bar{z}_N^k) \leq \Phi^{\hat{H}}(\bar{z}_1^k) \leq \Phi^{\hat{H}}(\mathbf{u}^k) \leq \Phi^{\hat{H}}(\mathbf{z}^k), \quad (4.10)$$

where the second inequality uses [Fact 4.1](#) and the last one is ensured by the linesearch condition in step 1.5. Moreover, in step 1.1, $z^{k+1} = \text{t}(\sum_{i=1}^N \nabla \hat{h}_i(\bar{z}_i^k))$, or equivalently stated by [Fact 4.2\(ii\)](#) $z^{k+1} = (z^{k+1}, \dots, z^{k+1}) \in \text{prox}_{\Phi}^{\hat{H}}(\bar{z}_N^k)$. Therefore, using [Fact 4.1](#) yields

$$\Phi^{\hat{H}}(\mathbf{z}^{k+1}) \leq \Phi^{\hat{H}}(\bar{z}_N^k) - D_{\hat{H}}(\mathbf{z}^{k+1}, \bar{z}_N^k). \quad (4.11)$$

Summing up the two inequalities in (4.10) and (4.11) yields

$$\Phi^{\hat{H}}(\mathbf{z}^{k+1}) \leq \Phi^{\hat{H}}(\mathbf{z}^k) - D_{\hat{H}}(\mathbf{z}^{k+1}, \bar{z}_N^k). \quad (4.12)$$

Noting that $\mathbf{z}^k = (z^k, \dots, z^k)$, the above inequality may be written as (cf. [Corollary 4.3](#))

$$\mathcal{L}(v^{k+1}, z^{k+1}) \leq \mathcal{L}(v^k, z^k) - \sum_{i=1}^N D_{\hat{h}_i}(z^{k+1}, \bar{z}_i^k).$$

[4.7\(ii\)](#): By reordering the inequality in (4.12) and telescoping we have

$$\sum_{k=0}^T D_{\hat{H}}(\mathbf{z}^{k+1}, \bar{z}_N^k) \leq \Phi^{\hat{H}}(\mathbf{z}^0) - \Phi^{\hat{H}}(\mathbf{z}^T) \leq \Phi^{\hat{H}}(\mathbf{z}^0) - \inf \Phi^{\hat{H}} < \infty.$$

The last two inequalities follow from the boundedness of $\Phi^{\hat{H}}$ from below, in light of [Assumption 1](#) and [Fact A.2\(iv\)](#). The inequality above shows that the sum is finite and hence $(D_{\hat{H}}(\mathbf{z}^{k+1}, \bar{\mathbf{z}}_N^k))_{k \in \mathbb{N}} \rightarrow 0$.

[4.7\(iii\)](#): The sequence $(\Phi^{\hat{H}}(\mathbf{z}^k))_{k \in \mathbb{N}} = (\mathcal{L}(v^k, z^k))_{k \in \mathbb{N}}$ is decreasing by [\(4.12\)](#) and since it is lower bounded, it should converge to a finite value φ_* with $\varphi(z^0) \geq \varphi_* \geq \inf \Phi^{\hat{H}} = \inf \Phi = \inf \varphi$. Moreover, from [\(4.11\)](#) and [\(4.10\)](#):

$$\Phi^{\hat{H}}(\mathbf{z}^{k+1}) + D_{\hat{H}}(\mathbf{z}^{k+1}, \bar{\mathbf{z}}_N^k) \leq \Phi^{\hat{H}}(\bar{\mathbf{z}}_N^k) \leq \Phi^{\hat{H}}(\mathbf{z}^k).$$

Since $D_{\hat{H}}(\mathbf{z}^{k+1}, \bar{\mathbf{z}}_N^k)$ vanishes, see [Theorem 4.7\(ii\)](#), $\Phi^{\hat{H}}(\bar{\mathbf{z}}_N^k) \rightarrow \varphi_*$, which in turn implies through the identity $\Phi^{\hat{H}}(\bar{\mathbf{z}}_N^k) = \Phi(\mathbf{z}^{k+1}) + D_{\hat{H}}(\mathbf{z}^{k+1}, \bar{\mathbf{z}}_N^k)$, that $(\Phi(\mathbf{z}^k))_{k \in \mathbb{N}} \rightarrow \varphi_*$.

[4.7\(iv\)](#): Take a subsequence $(\mathbf{z}_{k \in K}^k \rightarrow \mathbf{z}^*$ with $K \subseteq \mathbb{N}$. We have:

$$\varphi_* \stackrel{4.7(iii)}{\leftarrow} \frac{4.7(iii)}{k \in K} \Phi^{\hat{H}}(\mathbf{z}^k) - D_{\hat{H}}(\mathbf{z}^*, \mathbf{z}^k) \stackrel{A.2(iii)}{\leq} \Phi(\mathbf{z}^*) \stackrel{\text{lsc}}{\leq} \liminf_{k \in K} \Phi(\mathbf{z}^k) \stackrel{4.7(iii)}{=} \varphi_*. \quad (4.13)$$

[4.7\(v\)](#): Let $K \subseteq \mathbb{N}$ denote an infinite subsequence such that $(\mathbf{z}_{k \in K}^k \rightarrow \mathbf{z}^*$. It follows from [Theorem 4.7\(ii\)](#) along with [[63](#), Thm. 2.4] that $(\bar{\mathbf{z}}_N^k)_{k \in K} \rightarrow \mathbf{z}^*$. With $\mathbf{z}^{k+1} \in \text{prox}_{\Phi^{\hat{H}}}^{\hat{H}}(\bar{\mathbf{z}}_N^k)$ and the osc property of $\text{prox}_{\Phi^{\hat{H}}}^{\hat{H}}$ (see [Fact A.2\(i\)](#)), it follows that $\mathbf{z}^* \in \text{prox}_{\Phi^{\hat{H}}}^{\hat{H}}(\mathbf{z}^*)$ implying stationarity of the limit points as shown in [\(4.7\)](#).

[4.7\(vi\)](#): Level boundedness of φ implies that of $\Phi^{\hat{H}}$. It then follows from [\(4.11\)](#) and [\(4.10\)](#) that $(\bar{\mathbf{z}}_N^k)_{k \in \mathbb{N}}$ is contained in $\{\mathbf{w} : \Phi^{\hat{H}}(\mathbf{w}) \leq \Phi^{\hat{H}}(\bar{\mathbf{z}}_N^0)\}$, which is a bounded set. Boundedness of $(\mathbf{z}^k = (z^k, \dots, z^k))_{k \in \mathbb{N}}$ follows from that of $(\bar{\mathbf{z}}_N^k)_{k \in \mathbb{N}}$, local boundedness of the proximal mapping (see [Fact A.2\(i\)](#)), and $\mathbf{z}^{k+1} \in \text{prox}_{\Phi^{\hat{H}}}^{\hat{H}}(\bar{\mathbf{z}}_N^k)$. \square

4.4 Superlinear Convergence

In this section, we aim to demonstrate that the proposed linesearch is *smart*, particularly in identifying *mature* directions. When a direction d^k is deemed *mature*, the candidate update $\mathbf{z}^k + d^k$ will eventually be accepted without any backtracks, thereby enabling SPIRAL to exhibit superlinear convergence.

We first introduce necessary additional assumptions and lemmas. Subsequently, we delve into the identification of *mature* directions by SPIRAL and explore their relationship with quasi-Newton methods. Take the following assumptions:

Assumption 2 (superlinear convergence requirements). *The following hold in problem (1.1):*

A1 g is convex;

A2 for $i \in [N]$, $f_i, h_i \in \mathcal{C}^2$ with $\nabla^2 h_i \succ 0$.

To achieve superlinear convergence, SPIRAL requires that the sequence $(\mathbf{z}^k)_{k \in \mathbb{N}}$ converges to a strong local minimum \mathbf{z}^* of the cost function, and that the envelope is twice (strictly) differentiable at \mathbf{z}^* . It is noteworthy that the strong local minimality (isolated local minima) is a standard requirement for asymptotic properties of quasi-Newton methods. However, works such as [[2](#), [66](#)] relax this requirement to address nonisolated local minima as well. As future work, their techniques can be investigated for our setting.

To establish the aforementioned properties of the envelope, the following fact and lemma are presented, taking into account the additional [Assumption 2](#) in conjunction with [Assumption 1](#). First, let us formally define the concept of strong local minimality:

Definition 4.8 (Strong local minimum). *A point \mathbf{z}^* is said to be the strong local minimum of ϕ if there exist a neighborhood $\mathcal{N}_{\mathbf{z}^*}$ of \mathbf{z}^* and $c > 0$ such that for all $z \in \mathcal{N}_{\mathbf{z}^*}$, $\phi(z) \geq \phi(\mathbf{z}^*) + \frac{c}{2} \|z - \mathbf{z}^*\|^2$.*

The following fact establishes an equivalence between strong local minima of function φ and of its envelope $\varphi^{\hat{h}}$.

Fact 4.9 (equivalence of strong local minima [1, Thm. 3.7]). *Suppose that Assumptions 1 and 2 hold. Then, $z^* \in \text{fix prox}_{\varphi}^{\hat{h}}$ is a strong local minimum of φ if and only if it is a strong local minimum of $\varphi^{\hat{h}}$.*

We remark that $\text{prox}_{\varphi}^{\hat{h}}$ is single-valued whenever g is convex, hence all the fixed points are guaranteed to be *nondegenerate* in the sense of [1, Def. 3.5]. In order to achieve superlinear convergence, we assume z^* to be a strong local minimum of the cost φ , and that the envelope is twice (strictly) differentiable at this point. The subsequent lemma examines the second-order properties of the envelope to ensure the fulfillment of this requirement.

Lemma 4.10 (second order characterization). *Suppose that Assumptions 1 and 2 hold. Then, given $z^* \in \text{fix prox}_{\varphi}^{\hat{h}}$, there exists a neighborhood of z^* where $r_{\hat{h}}$ is Lipschitz continuous, and $\varphi^{\hat{h}}$ is continuously differentiable. If, in addition, t is (strictly) differentiable at $\nabla \hat{h}(z^*)$, then*

(i) $r_{\hat{h}}$ is (strictly) differentiable at z^* with $Jr_{\hat{h}}(z^*) = I - Jt(\nabla \hat{h}(z^*))\nabla^2 \hat{h}(z^*)$;

(ii) $\varphi^{\hat{h}}$ is twice (strictly) differentiable at z^* with symmetric Hessian

$$\nabla^2 \varphi^{\hat{h}}(z^*) = \nabla^2 \hat{h}(z^*) Jr_{\hat{h}}(z^*).$$

In particular, if z^ is a strong local minimum of φ , then $\nabla^2 \varphi^{\hat{h}}(z^*)$ is symmetric positive definite and $Jr_{\hat{h}}(z^*)$ is invertible.*

Proof. Observe that $t_{\hat{h}} = t \circ \nabla \hat{h} = \text{prox}_{\varphi}^{\hat{h}}$ as shown in Corollary 4.3. Given this characterization, the first claim follows directly from [1, Thm. 3.10].

4.10(i) Since $\nabla \hat{h} \in C^1$ (cf. Assumption 2.A2) and t is (strictly) differentiable at $\nabla \hat{h}(z^*)$, so is the composition $t \circ \nabla \hat{h}$, thus implying (strict) differentiability of $r_{\hat{h}} = \text{id} - t_{\hat{h}}$. The Jacobian of the residual is obtained by the chain rule.

4.10(ii): The claim follows from (strict) differentiability of t at $\nabla \hat{h}(z^*)$. Moreover, $\nabla^2 \hat{h} \succ 0$ owing to Assumptions 1.A1 and 2.A2 and $\gamma_i \in (0, N/L_i)$. Thus, $Jr_{\hat{h}} z^*$ is nonsingular when z^* is a strong local minimum of $\varphi^{\hat{h}}$, or, equivalently, of φ (cf. Fact 4.9). \square

After establishing the desirable properties of the envelope in the vicinity of fixed point $z^* \in \text{fix prox}_{\varphi}^{\hat{h}}$, we now proceed to characterize the quality of directions d^k in step 1.4 of Algorithm 1 through introducing the notion of *superlinear directions* which was introduced in the seminal work [27]. We also refer the reader to the works such as [66, 1] for further discussion and extensions.

Definition 4.11 (superlinear directions). *Relative to a sequence $(z^k)_{k \in \mathbb{N}}$ that converges to a point $z^* \in \mathbb{R}^n$, we say that $(d^k)_{k \in \mathbb{N}} \subseteq \mathbb{R}^n$ is a sequence of superlinear directions, if*

$$\lim_{k \rightarrow \infty} \frac{\|z^k + d^k - z^*\|}{\|z^k - z^*\|} = 0.$$

The following theorem provides asymptotic guarantees for the superlinear convergence of SPIRAL. As remarked before, when the directions satisfy Definition 4.11, the backtracking linesearch will eventually never be triggered, thus substantially improving the performance of the algorithm. The theorem requires local assumptions such as strict differentiability of t in (3.1) at $\nabla \hat{h}(z^*)$, where z^* is the limit point of $(z^k)_{k \in \mathbb{N}}$. Auxiliary results for controlling the terms $\|\tilde{z}_{\ell}^k - u^k\|$ for $\ell \in [N]$ that appear due to the (shuffled) cyclic updates are postponed to Lemma B.1 in Appendix B.

Theorem 4.12 (superlinear convergence). *Consider the sequence $(z^k)_{k \in \mathbb{N}}$ generated by Algorithm 1, and additionally to Assumptions 1 and 2, suppose the following are satisfied:*

A1 $(z^k)_{k \in \mathbb{N}}$ converges to a strong local minimum z^* of φ ;

A2 the directions d^k are superlinear relative to $(z^k)_{k \in \mathbb{N}}$ (cf. [Definition 4.11](#));

A3 \mathfrak{t} (defined in [\(3.1\)](#)) is strictly differentiable at $\nabla \hat{h}(z^*)$.

Then, asymptotically the linesearch in [step 1.5](#) will be accepted with $\tau = 1$, and $(z^k)_{k \in \mathbb{N}}$ converges to z^* at superlinear rate.

Proof. Let $u_0^k := z^k + d^k$. Due to the superlinearity of the directions d^k ,

$$\lim_{k \rightarrow \infty} \frac{\|u_0^k - z^*\|}{\|z^k - z^*\|} = 0. \quad (4.14)$$

We start by showing that close enough to the limit point the linesearch condition would always be satisfied with $\tau = 1$. It follows from [Theorem 4.7\(v\)](#) that $z^* \in \text{fix prox}_\varphi^{\hat{h}}$ and from [Fact 4.9](#) that z^* is also a strong local minimum of $\varphi^{\hat{h}}$. Hence, $G^* = \nabla^2 \varphi^{\hat{h}}(z^*)$ is symmetric positive definite by [Lemma 4.10\(ii\)](#). Let

$$\varepsilon_k := \frac{\varphi^{\hat{h}}(u_0^k) - \varphi_*}{\varphi^{\hat{h}}(z^k) - \varphi_*}.$$

Since $G^* := \nabla^2 \varphi^{\hat{h}}(z^*) \succ 0$, a second-order expansion of $\varphi^{\hat{h}}$ at z^* yields

$$\begin{aligned} \lim_{k \rightarrow \infty} \varepsilon_k &= \lim_{k \rightarrow \infty} \frac{\frac{1}{2} \langle G^*(u_0^k - z^*), u_0^k - z^* \rangle + \mathcal{O}(\|u_0^k - z^*\|^2)}{\frac{1}{2} \langle G^*(z^k - z^*), z^k - z^* \rangle + \mathcal{O}(\|z^k - z^*\|^2)} \\ &\leq \lim_{k \rightarrow \infty} \frac{\|G^*\| \|u_0^k - z^*\|^2 + \mathcal{O}(\|u_0^k - z^*\|^2)}{\lambda_{\min}(G^*) \|z^k - z^*\|^2 + \mathcal{O}(\|z^k - z^*\|^2)} \\ &= \lim_{k \rightarrow \infty} \frac{\|G^*\| \frac{\|u_0^k - z^*\|^2}{\|z^k - z^*\|^2} + \frac{\mathcal{O}(\|u_0^k - z^*\|^2)}{\|z^k - z^*\|^2}}{\lambda_{\min}(G^*) + \frac{\mathcal{O}(\|z^k - z^*\|^2)}{\|z^k - z^*\|^2}} \stackrel{(4.14)}{\leq} 0. \end{aligned}$$

In particular, there exists $k_0 \in \mathbb{N}$ such that $\varepsilon_k \leq 1 \ \forall k \geq k_0$. Moreover, since z^k converges to z^* , it follows from [Theorem 4.7\(i\)](#) and [Corollary 4.3](#) that $\varphi^{\hat{h}}(z^k) \geq \varphi^{\hat{h}}(z^*)$. Consequently, using the definition of ε_k above with $\varphi_* = \varphi^{\hat{h}}(z^*)$ due to [Fact A.2\(iv\)](#),

$$\varphi^{\hat{h}}(u_0^k) - \varphi^{\hat{h}}(z^k) = -(1 - \varepsilon_k) \left(\varphi^{\hat{h}}(z^k) - \varphi^{\hat{h}}(z^*) \right) \leq 0 \quad \forall k \geq k_0.$$

Therefore, the unit stepsize would always be accepted in [step 1.5](#). Now, with unit stepsize $u_0^k = u^k$, as in [step 1.5.a](#), and this implies through [\(4.14\)](#) that

$$\lim_{k \rightarrow \infty} \frac{\|u^k - z^*\|}{\|z^k - z^*\|} = 0. \quad (4.15)$$

On the other hand

$$\begin{aligned} \|\bar{z}_N^k - \mathbf{u}^k\| &\stackrel{(4.8)}{\leq} \sum_{i=1}^N \|\bar{z}_i^k - u^k\| \stackrel{(B.1)}{\leq} \eta \|\bar{z}_1^k - u^k\| \\ &\text{triangular inequality} \leq \eta \|\bar{z}_1^k - z^*\| + \eta \|u^k - z^*\| \\ \text{step 2.7 of Algorithm 2} &= \frac{\eta}{\sqrt{N}} \|\bar{z}_1^k - z^*\| + \eta \|u^k - z^*\| \\ &\leq \eta(\bar{L} + 1) \|u^k - z^*\|, \end{aligned} \quad (4.16)$$

where $\eta = \sum_{i=1}^N c_i$ and the last inequality follows from local Lipschitz continuity of $\text{prox}_{\Phi}^{\hat{H}}$ and step 2.5 of Algorithm 2. Further exploiting local Lipschitz continuity of the proximal mapping

$$\begin{aligned} \|z^{k+1} - z^*\| &= \frac{1}{\sqrt{N}} \|z^{k+1} - z^*\| \\ \text{Lip. cont. of } \text{prox}_{\Phi}^{\hat{H}} \text{ and step 2.1 of Algorithm 2} &\leq \frac{\bar{L}}{\sqrt{N}} \|\bar{z}_N^k - z^*\| \\ \text{triangular inequality} &\leq \frac{\bar{L}}{\sqrt{N}} \|\bar{z}_N^k - \mathbf{u}^k\| + \frac{\bar{L}}{\sqrt{N}} \|\mathbf{u}^k - z^*\| \\ (4.16) &\leq \alpha \|u^k - z^*\| \end{aligned}$$

where $\alpha = \frac{\eta}{\sqrt{N}} \bar{L}(\bar{L} + 1) + \bar{L}$. Hence, combined with (4.15)

$$\frac{\|z^{k+1} - z^*\|}{\|z^k - z^*\|} \leq \alpha \frac{\|u^k - z^*\|}{\|z^k - z^*\|} \rightarrow 0,$$

establishing the claimed superlinear convergence. \square

A well-known condition for analyzing quasi-Newton methods is the celebrated Dennis-Moré condition [23, 24], which characterizes the quality of the directions as follows:

$$\lim_{k \rightarrow \infty} \frac{\|r_{\hat{h}}(z^k) + \text{J}r_{\hat{h}}(z^*)d^k\|}{\|d^k\|} = 0. \quad (4.17)$$

This classical condition, in conjunction with Assumptions 1 and 2 and Theorems 4.12.A1 and 4.12.A3, leads to the emergence of superlinear directions [1, Thm. 5.13].

Note that the directions computed by Broyden updates, as one of the quasi-Newton methods, provably satisfy the Dennis-Moré condition stated in (4.17) (refer to [68, Thm. 5.11] and [66, Thm. VI.8]). In order to achieve this, Broyden updates require the aforementioned regularity conditions on $r_{\hat{h}}$ at z^* , as well as the boundedness of low-rank updates H^k in (3.5). It is important to mention that, although it is not formally established that L-BFGS satisfies the Dennis-Moré condition, L-BFGS performs better than Broyden updates in practical scenarios. The theoretical examination of the Dennis-Moré condition using L-BFGS updates is considered as a future research direction.

It is noteworthy that the boundedness of low-rank updates H^k —similarly existence of the bound $\|d^k\| \leq D\|z^k - v^k\|$ with some finite $D \geq \sup_{k \in \mathbb{N}} \|H^k\|$ due to (3.5), as it will be required in Theorem 4.16—is a common assumption in the analysis of quasi-Newton methods (see, e.g., [66, Ass. 2] and [1, Thm. 5.7-A3 and Thm. 5.8-A3]), and is guaranteed by employing safeguards in practice.

Remark 4.13 (practical considerations). The condition $\|d^k\| \leq D\|z^k - v^k\|$ is mild in practice since, as a safeguard here, in the case of failure in meeting the inequality, the directions may be scaled by $d^k \leftarrow D \frac{\|z^k - v^k\|}{\|d^k\|} d^k$ using a sufficiently large predefined scalar D . Moreover, failure in meeting the inequality does not deteriorate the global and subsequential convergence of SPIRAL, as long as $\|d^k\|$ is scaled whenever necessary, as demonstrated in Theorem 4.7. As a final remark, although failure is possible (especially in the initial iterations when d^k is not *mature*), it has not occurred in any of our simulations in Section 5.

Remark 4.14. We remark that global convergence results in Theorem 4.7 is established for any choice of direction. Even though in Algorithm 1 quasi-Newton directions based on the residual mapping were suggested (cf. (3.5)), any superlinear direction can be employed in the algorithm. As a result, our theory provides a direct globalization strategy for works that employ quasi-Newton direction with only local convergence guarantees. For instance, it globalizes the recent work [45] which studies smooth and strongly convex finite sum problems, and proposes an incremental quasi-Newton method with local convergence guarantees.

4.5 Sequential and Linear Convergence

In accordance with [Theorem 4.12](#) presented in the previous subsection, in order to achieve superlinear convergence, SPIRAL requires to have a sequence $(z^k)_{k \in \mathbb{N}}$ that converges to a strong local minimum of the cost function φ . The subsequent theorem establishes conditions under which the entire sequence $(z^k)_{k \in \mathbb{N}}$ converges to a stationary point with a linear convergence rate. To accomplish this, an additional assumption is required, namely, the Kurdyka-Łojasiewicz (KL) property of the full cost function [\[38\]](#). It is worth noting that φ possesses the KL property for a wide range of problems, including situations where f_i and g are semialgebraic functions, which are commonly encountered in various applications (refer to [\[9, 10\]](#) for further elaboration). The formal statement of the KL property is as follows:

Definition 4.15. (*KL property with exponent θ*) A proper lsc function $\phi : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$ has the Kurdyka-Łojasiewicz (KL) property with exponent $\theta \in (0, 1)$ if for every $z^* \in \text{dom } \partial\phi$ there exist constants $\eta, \epsilon, \rho > 0$ such that

$$\psi'(\phi(z) - \phi(z^*)) \text{dist}(0, \partial\phi(z)) \geq 1, \quad \psi(s) := \rho s^{1-\theta}, \quad (4.18)$$

for all z such that $\|z - z^*\| \leq \epsilon$ and $\phi(z^*) < \phi(z) < \phi(z^*) + \eta$.

In the following result, we present convergence guarantees for *sequential* convergence under the KL assumption on the cost function. The proof follows standard techniques found in the literature, drawing inspiration from works such as [\[67, 1, 39\]](#). For the sake of completeness, we provide the proof in [Appendix C](#). The primary challenge in establishing the *sequential* convergence of [Algorithm 1](#) lies in demonstrating the bound $\|z^{k+1} - z^k\| \leq C\|z^k - z_N^{k-1}\|$ for a positive constant C , as required by [Theorem 4.16](#). This bound is established by [Lemma B.2](#) presented in [Appendix B](#).

Theorem 4.16 (Sequential and linear convergence). *Additionally to [Assumptions 1 and 2](#), suppose the following is satisfied:*

- A1 φ is level bounded;
- A2 φ has the KL property (cf. [Definition 4.15](#)) with exponent $\theta \in (0, 1)$;
- A3 the directions d^k in [step 1.4](#) satisfy $\|d^k\| \leq D\|z^k - v^k\|$ for some $D \geq 0$.

Then, $(z^k)_{k \in \mathbb{N}}$ converges to a stationary point z^* for φ . Moreover, if the KL function has the exponent parameter in the range $\theta \in (0, 1/2]$, then $(z^k)_{k \in \mathbb{N}}$ and $(\varphi(z^k))_{k \in \mathbb{N}}$ converge at R -linear rate.

Proof. Refer to [Appendix C](#). □

5 Numerical Experiments

In this section, we evaluate the proposed algorithm, SPIRAL, for both convex and nonconvex problems, considering cost functions with and without Lipschitz continuous gradients. We examine two versions of SPIRAL: 1) SPIRAL, which follows [Algorithm 1](#), and 2) adaSPIRAL, an adaptive version with additional steps as outlined in [Table 1](#). We compare SPIRAL against proxSARAH [\[53\]](#), proxSVRG [\[56\]](#), proxSGD [\[31\]](#), proxSAGA [\[56\]](#), Finito/MISO [\[22, 43\]](#), and low-memory Finito/MISO [\[39, Alg. 2\]](#). For the convex ℓ_1 regularized least squares problem, we compare against Finito/MISO [\[22\]](#). For the nonconvex nonnegative principal component analysis problem, we compare against [\[40\]](#), which addresses Finito/MISO in the general nonsmooth nonconvex case. Additionally, we compare SPIRAL against SMD and the Bregman Finito/MISO method [\[39\]](#) for the phase retrieval problem, where the cost function lacks a Lipschitz continuous gradient. The databases used in the evaluations are from LIBSVM [\[17\]](#). To assess the performance of the algorithms, we employ the suboptimality criterion

$$\mathcal{D}(z^k) := \|z^k - v^k\| \quad (5.1)$$

with $v^k \in \mathfrak{t}_{\hat{h}}(z^k)$, since

$$\begin{aligned} \text{dist}(0, \hat{\partial}\varphi(v^k)) &\leq \inf_{v^k \in \mathfrak{t}_{\hat{h}}(z^k)} \left\| \sum_{i=1}^N \nabla \hat{h}_i(z^k) - \nabla \hat{h}_i(v^k) \right\| \\ &\leq \sum_{i=1}^N \|\nabla \hat{h}_i(z^k) - \nabla \hat{h}_i(v^k)\| \leq c \|z^k - v^k\|, \end{aligned}$$

where the first inequality holds by [Corollary 4.3\(ii\)](#) in [Section 4.1](#), and $c > 0$ is some constant due to local Lipschitz continuity of \hat{h}_i , and the fact that z^k and v^k remain bounded.

For all the algorithms in the comparisons, the stepsizes are set according to their theoretical convergence studies. Refer to [\[53, Thm. 8\]](#) for proxSARAH, [\[56, Thm. 1\]](#) for proxSVRG, [\[56, Thm. 3\]](#) for nonconvex proxSAGA, and [\[21\]](#) for convex proxSAGA. For proxSGD the diminishing stepsize $\gamma^t = \frac{\gamma_0}{1+\tilde{\gamma}t}$ is considered according to [\[30\]](#) with t as the epoch counter, $\gamma_0 = 0.1$, and $\tilde{\gamma} = 0.5$. For (Bregman) Finito/MISO and also all the SPIRAL versions the stepsizes are set $\gamma_i = \frac{\alpha N}{L_i}$, $i \in [N]$ with $\alpha = 0.999$. For adaSPIRAL, the stepsizes are all initialized by $\kappa \times \max_{i \in [N]} \{\frac{N}{L_i}\}$, with a grid search for $\kappa \in \{5, 10, 50, 100\}$ for each plot. Furthermore, the quasi-Newton directions in [step 1.4](#) are computed using [\(3.5\)](#), where H^k is updated using the L-BFGS method with a memory size of 5. The maximum number of backtracks q_{\max} is also set equal to 5. It should be mentioned that while directions in [step 1.4](#) can be computed using any quasi-Newton method (e.g. Broyden updates as discussed in [Section 4.4](#)), L-BFGS yields superior numerical results. Finally, we refer to epochs by counting the total number of individual gradient evaluations divided by N , including those involved in the linesearches.

5.1 Adaptive variant

Using the global smoothness constants may lead to conservative stepsizes. [Table 1](#) expands [steps 1.1, 1.3, 1.5, and 1.9](#) of [Algorithm 1](#) in order to clarify the order of operations with the added backtracking linesearches that ensure the fundamental descent property in [Fact 2.4](#). This adaptation allows SPIRAL to estimate (relative) smoothness moduli locally—as opposed to using global estimates—resulting in larger stepsizes. The reader is referred to [Appendix E.1](#) for further explanations about the memory-efficient implementation of the adaptive variant.

5.2 Sparse Phase Retrieval with Squared Loss

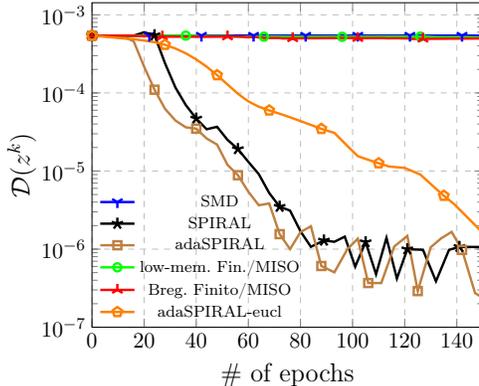
In the initial set of simulations, we evaluate the performance of SPIRAL on the sparse phase retrieval problem, which involves signal recovery based on intensity measurements. This problem finds applications in various fields, such as electron microscopy, speech recognition, optical imaging, and X-ray crystallography [\[64, 15\]](#). The sparse phase retrieval problem is formulated as follows:

$$\text{minimize}_{z \in \mathbb{R}^n} \frac{1}{N} \sum_{i=1}^N \text{Loss}(b_i, \langle a_i, z \rangle^2) + g(z), \quad (5.2)$$

where $a_i \in \mathbb{R}^n$ and $b_i \in \mathbb{R}_+$ are data vectors. The objective of this optimization problem is to find a sparse vector $z \in \mathbb{R}^n$ that best approximates b_i as $\langle a_i, z \rangle^2$ for all $i \in [N]$. Sparsity is introduced to account for noise and outliers, and this can be achieved by selecting nonsmooth functions such as the ℓ_0 or ℓ_1 norms. In this simulation, we consider the squared loss function $\text{Loss}(x, y) = \frac{1}{4}(x - y)^2$ and $g = \lambda \|\cdot\|_1$. To cast [\(5.2\)](#) into the original optimization problem form [\(1.1\)](#), we define the functions as follows:

$$f_i(z) = \frac{1}{4}(\langle a_i, z \rangle^2 - b_i)^2, \text{ with } h_i(z) = h(z) = \frac{1}{4}\|z\|^4 + \frac{1}{2}\|z\|^2.$$

Although the cost functions f_i do not possess Lipschitz continuous gradients, they exhibit smoothness relative to the reference function h [\[12, Lem. 5.1, Prop. 5.1 and 5.2\]](#).



(a) Performance of different algorithms.



(b) Image recovery of adaSPIRAL.

Figure 1: Performance for the phase retrieval problem (5.2) on a digit 6 image with $N = 1280$, $n = 256$. Image recovery is after 100 epochs, including the original image (left), initialization (center), and output (right).

In this simulation, we consider 16×16 gray-scale images of digits from the dataset [35]. The images are vectorized, so $n = 256$. The matrix $A \in \mathbb{R}^{N \times n}$ with a_i being its i th row, with $N = d \times n$ and $d = 5$ is generated according to the procedure described in [26]. We form this matrix as $A = [MS_1, \dots, MS_N]$ with $M \in \mathbb{R}^{n \times n}$ a normalized Hadamard matrix and S_i diagonal sign matrices with the diagonal elements in $\{-1, 1\}$. For noiseless data, $d = 3$ is sufficient for a complete recovery. The measurements are corrupted by setting $b_i = 0$ with probability $p_c = 0.02$. Also, we set $\lambda = \frac{1}{N}$ by the hyperparameter search to have a visually good solution. Furthermore, the algorithms are initialized with the initialization scheme suggested in [26], and they converge to the same local optimal point. The performance of different algorithms is shown in Figure 1a for a digit 6 image.

As depicted in the figure, SPIRAL demonstrates significantly faster performance compared to the other algorithms. Even though the cost function in this scenario does not possess a Lipschitz continuous gradient, we evaluate the performance of adaSPIRAL, both in Bregman and Euclidean versions, in Figure 1a. It is important to note that adaSPIRAL-eucl, implemented according to Algorithm 3 with the additional steps outlined in Table 1 using dgfs $h_i = \frac{1}{2} \|\cdot\|^2$, does not require any prior knowledge of Lipschitz constants L_i . Remarkably, adaSPIRAL-eucl performs well on cost functions without Lipschitz continuous gradients, as verified by this simulation. Consequently, adaSPIRAL-eucl demonstrates potential applicability to a wider range of cost functions in various applications. Additionally, adaSPIRAL outperforms SPIRAL due to its ability to employ larger stepsizes that are dynamically updated as needed, thereby speeding up convergence. Furthermore, as shown in Figure 1b, adaSPIRAL-eucl exhibits good image recovery capabilities even in highly corrupted initial conditions.

5.3 ℓ_1 Regularized Least Squares Problem

In this section, we evaluate the performance of SPIRAL for the Lasso problem, which is a convex optimization problem commonly used for regression tasks. The Lasso formulation is given by

$$\text{minimize}_{z \in \mathbb{R}^n} \frac{1}{2} \|Az - b\|_2^2 + \lambda \|z\|_1 \quad (5.3)$$

where A is a matrix with data vectors $a_i \in \mathbb{R}^n$ as its rows, and b is a vector with corresponding labels $b_i \in \mathbb{R}$. The datasets used for regression tasks include the *mg*, *cadata*, *housing*, and *triazines* datasets obtained from LIBSVM. Additionally, synthetic datasets are generated using the procedure described in [49, §6] for two different dimensions. The parameter λ is appropriately set for each dataset.

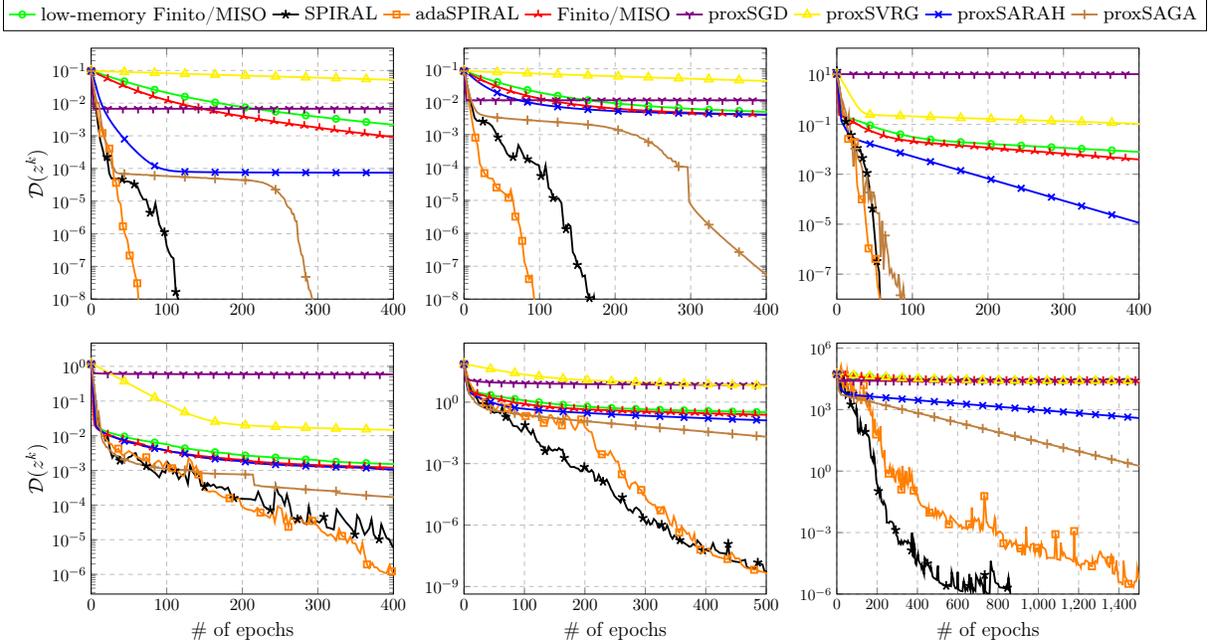


Figure 2: Performance of different algorithms for the Lasso problem (5.3). Synthetic dataset (top left) with $N = 10000$, $n = 400$, synthetic dataset (top center) with $N = 300$, $n = 600$, mg (top right) with $N = 1385$, $n = 6$, triazines (bottom left) with $N = 186$, $n = 60$, housing (bottom center) with $N = 506$, $n = 13$, and cadata (bottom right) with $N = 20640$, $n = 8$.

Figure 2 provides a comparison of different algorithms on six datasets. It is evident from the results that both SPIRAL and adaSPIRAL exhibit superior convergence performance compared to other algorithms, regardless of whether the datasets are synthetic or practical. Also the same speed up by adaSPIRAL is evident for most of the datasets. Note that adaSPIRAL does not require a priori knowledge of Lipschitz constants L_i , and still its performance is comparable with that of SPIRAL. Compared to Finito/MISO, low-memory Finito/MISO is worse, however, it does not need a large memory to store the gradient vectors. It is also observed that SPIRAL is particularly fast on dense datasets.

5.4 Nonnegative Principal Component Analysis

In this section, we investigate the problem of nonnegative principal component analysis (NN-PCA), which has also been studied in previous works [56, 53]. The problem is formulated as follows:

$$\text{minimize}_{z \in \mathbb{R}^n} f(z) := -\frac{1}{2N} \sum_{i=1}^N z^T (a_i a_i^T) z \quad \text{subject to } \|z\| \leq 1, z \geq 0, \quad (5.4)$$

where N denotes the number of data points represented by $a_i \in \mathbb{R}^n$. To cast the problem with the form of (1.1), we define $f_i(z) = -\frac{1}{2} z^T (a_i a_i^T) z$ and $g(z) = \delta_{\mathcal{B}}(z)$, where $\mathcal{B} := \{w \in \mathbb{R}^n \mid \|w\| \leq 1, w \geq 0\}$ represents the constraints of the NN-PCA problem (5.4). It is worth mentioning that SPIRAL allows for different smoothness constants and individualized stepsizes γ_i for each of the functions f_i . In this case, the data points a_i are not normalized to improve the output of NN-PCA analysis, providing a better representation of the dataset. All the algorithms employed for this nonconvex optimization problem are initialized with running proxSGD for 10 epochs, starting from the same initial point, ensuring convergence to a similar local optimum. The optimality criterion (5.1) is reported as a function of the number of epochs.

As depicted in Figure 3, the quasi-Newton updates in SPIRAL significantly enhance the convergence rate compared to (low-memory) Finito/MISO, which lacks such updates. Although proxSARAH exhibits

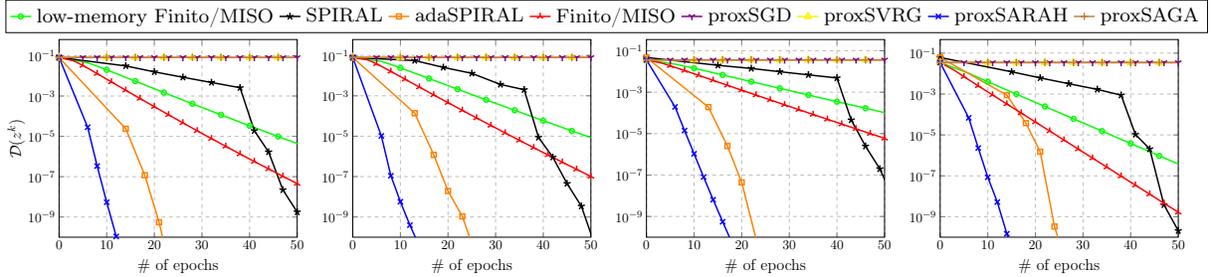


Figure 3: Performance of different algorithms for the NN-PCA problem of (5.4). MNIST (left) with $N = 60000$, $n = 784$, covtype (left center) with $N = 581012$, $n = 54$, a9a (right center) with $N = 32561$, $n = 123$, and aloi (right) with $N = 108000$, $n = 128$.

faster convergence for this problem, it performs slower in the Lasso problem and is unable to handle non-Lipschitz differentiable cost functions, as demonstrated in the problem discussed in Section 5.2.

In order to demonstrate how competitive different versions of SPIRAL are with state-of-the-art methods used in ML, also in terms of CPU time, performance comparisons are conducted in Appendix D. It is worth noting that despite SPIRAL’s approximation of second-order information, the adopted quasi-Newton method, namely L-BFGS, demonstrates efficiency by relying solely on inexpensive level 1 BLAS operations, such as inner products, scalar multiplications, and additions.

6 Conclusion

This paper introduced SPIRAL, an optimization algorithm designed for solving regularized finite sum minimization problems. SPIRAL operates in a nonconvex setting and does not rely on the typical assumption of Lipschitz differentiability. Many existing methods that utilize quasi-Newton directions in finite sum settings either impose restrictive conditions or only achieve local convergence. In contrast, we demonstrated that SPIRAL achieves a superlinear convergence rate while ensuring *global* convergence, all without the need for diminishing step sizes, and under standard mild assumptions. This is achieved by the introduction of a straightforward yet effective linesearch which is *smart*, in the sense that it will never be triggered close enough to (sufficiently regular) solutions—an aspect validated also in our simulations. Moreover, it is observed that while addressing nonsmooth nonconvex problems, SPIRAL is still competitive with the state-of-the-art on classical convex problems, such as regularized least squares problems. Promising future research directions include the adaptation of SPIRAL to domains like distributed and federated learning.

A Preliminaries

Fact A.1 (basic properties [18, 50]). *The following hold for a dgf $H : \mathbb{R}^n \rightarrow \mathbb{R}$, $x, y, z \in \mathbb{R}^n$:*

- (i) (three-point inequality) $D_H(x, z) = D_H(x, y) + D_H(y, z) + \langle x - y, \nabla H(y) - \nabla H(z) \rangle$. [18, Lem. 3.1].

For any convex set $\mathcal{U} \subseteq \mathbb{R}^n$ and $u, v \in \mathcal{U}$ the following hold [50, Thm. 2.1.5, 2.1.10]:

- (ii) *If H is $\mu_{H, \mathcal{U}}$ -strongly convex on \mathcal{U} , then $\frac{\mu_{H, \mathcal{U}}}{2} \|v - u\|^2 \leq D_H(v, u) \leq \frac{1}{2\mu_{H, \mathcal{U}}} \|\nabla H(v) - \nabla H(u)\|^2$.*
- (iii) *If ∇H is $\ell_{H, \mathcal{U}}$ -Lipschitz on \mathcal{U} , then $\frac{1}{2\ell_{H, \mathcal{U}}} \|\nabla H(v) - \nabla H(u)\|^2 \leq D_H(v, u) \leq \frac{\ell_{H, \mathcal{U}}}{2} \|v - u\|^2$.*

In the following, some properties of the Bregman Moreau envelope are highlighted. The interested reader is referred to [1] and [37] for proofs and further properties.

Fact A.2 (Basic properties of ϕ^H and prox_ϕ^H , [1, 37]). Let $H : \mathbb{R}^n \rightarrow \mathbb{R}$ denote a dgf (cf. Definition 2.1), and $\phi : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ be a proper lsc, and lower bounded function. Then, the following hold:

- (i) prox_ϕ^H is locally bounded, compact-valued, and outer semicontinuous;
- (ii) ϕ^H is finite-valued and continuous; it is locally Lipschitz if so is ∇H ;
- (iii) $\phi^H(z) = \phi(v) + D_H(v, z) \leq \phi(y) + D_H(y, z)$ with any $y, z \in \mathbb{R}^n$, $v \in \text{prox}_\phi^H(z)$. Hence, $\phi^H(z) \leq \phi(z)$;
- (iv) $\inf \phi = \inf \phi^H$ and $\text{argmin } \phi^H = \text{argmin } \phi$;
- (v) ϕ^H is level-bounded iff so is ϕ .

The following fact studies sufficient conditions for Lipschitz continuity of the Bregman proximal mapping and continuity of the Moreau envelope, both of which are crucial to the theory developed in Theorems 4.7 and 4.12.

Fact A.3 ([39, Lem. A.2]). Let $\mathcal{V}_i \subseteq \mathbb{R}^n$ be nonempty and convex, $i \in [N]$, and let $\mathcal{V} := \mathcal{V}_1 \times \dots \times \mathcal{V}_N$. Additionally to Assumption 1, suppose that g is convex, and h_i , $i \in [N]$, is ℓ_{h_i} -smooth and μ_{h_i} -strongly convex on \mathcal{V}_i . Then, the following hold for function \hat{H} as in (4.2) with $\gamma_i \in (0, N/L_{f_i})$, $i \in [N]$:

- (i) $\text{prox}_\Phi^{\hat{H}}$ is \bar{L} -Lipschitz continuous on \mathcal{V} for some constant $\bar{L} \geq 0$.

If in addition f_i and h_i are twice continuously differentiable on \mathcal{V}_i , $i \in [N]$, then

- (ii) $\Phi^{\hat{H}}$ is continuously differentiable on \mathcal{V} with $\nabla \Phi^{\hat{H}} = \nabla^2 \hat{H} \circ (\text{id} - \text{prox}_\Phi^{\hat{H}})$.

The following fact establishes the equivalence between problems (1.1) and (4.1).

Fact A.4 ([39, Lem. A.1]). Let the functions φ and Φ be as in (1.1) and (4.1), respectively. Then,

- (i) $\partial \Phi(\mathbf{x}) = \{v = (v, \dots, v) \mid \sum_i v_i \in \partial \varphi(x)\}$ if $\mathbf{x} = (x, \dots, x) \in \Delta$, and is empty otherwise.
- (ii) Φ has the KL property at $\mathbf{x} = (x, \dots, x)$ iff so does φ at x . In this case, the desingularizing functions are the same up to a positive scaling.

B Omitted lemmas

Lemma B.1. Suppose that Assumptions 1 and 2 hold and that φ is level bounded. Consider the sequence generated by Algorithm 1. Then, for every $\ell \in [N]$ there exists $c_\ell > 0$ such that

$$\|\tilde{z}_\ell^k - u^k\| \leq c_\ell \|\tilde{z}_1^k - u^k\|. \quad (\text{B.1})$$

Proof. By level boundedness of φ and Theorem 4.7, $(\mathbf{u}^k)_{k \in \mathbb{N}}$, $(\mathbf{z}^k)_{k \in \mathbb{N}}$, $(\tilde{z}_i^k)_{k \in \mathbb{N}}$, $i \in [N]$ are contained in a nonempty bounded set \mathcal{U} . By Assumption 2.A2, h_i is locally strongly convex and locally Lipschitz, which along with Assumption 2.A1 and Fact A.3 implies that $\text{prox}_\Phi^{\hat{H}}$ is \bar{L} -Lipschitz on a convex subset of \mathcal{U} for some $\bar{L} > 0$. Without loss of generality and for the sake of simplicity, we assume the cyclic sweeping rule in the incremental loop, i.e., $i^\ell = \ell$. Note that the following proof can be easily cast into the case of cyclic sweeping without replacement. Arguing by induction, for $\ell = 1$, (B.1) holds trivially. Suppose that the claim holds for some $\ell \geq 1$. Then, by triangular inequality and the definition of \tilde{z}_ℓ^k in step 2.7 of

Algorithm 2

$$\begin{aligned}
\|\tilde{z}_{\ell+1}^k - \mathbf{u}^k\| &= \frac{1}{\sqrt{N}} \|\tilde{z}_{\ell+1}^k - \mathbf{u}^k\| \\
&\leq \frac{1}{\sqrt{N}} \|\tilde{z}_1^k - \mathbf{u}^k\| + \frac{1}{\sqrt{N}} \|\tilde{z}_{\ell+1}^k - \tilde{z}_1^k\| \\
\text{Lip. continuity of } \text{prox}_{\Phi}^{\hat{H}} \text{ and Algorithm 2} &\leq \frac{1}{\sqrt{N}} \|\tilde{z}_1^k - \mathbf{u}^k\| + \frac{\bar{L}}{\sqrt{N}} \|\tilde{z}_{\ell+1}^k - \mathbf{u}^k\| \\
(4.8) &\leq \|\tilde{z}_1^k - \mathbf{u}^k\| + \frac{\bar{L}}{\sqrt{N}} \sum_{j \leq \ell} \|\tilde{z}_j^k - \mathbf{u}^k\| \\
(\text{induction}) &\leq \underbrace{\left(1 + \frac{\bar{L}}{\sqrt{N}} \sum_{j \leq \ell} c_j\right)}_{:= c_{\ell+1}} \|\tilde{z}_1^k - \mathbf{u}^k\|,
\end{aligned}$$

establishing (B.1). \square

Lemma B.2. *In addition to the assumptions in Lemma B.1, suppose that the directions d^k in step 1.4 satisfy $\|d^k\| \leq D\|z^k - v^k\|$ for some $D \geq 0$. Then, $\|z^{k+1} - z^k\| \leq C\|z^k - \bar{z}_N^{k-1}\|$ holds for some positive C .*

Proof. By the same reasoning as in Lemma B.1, $\text{prox}_{\Phi}^{\hat{H}}$ is \bar{L} -Lipschitz continuous on a bounded convex set containing the iterates $(\mathbf{u}^k)_{k \in \mathbb{N}}$, $(z^k)_{k \in \mathbb{N}}$, $(\tilde{z}_i^k)_{k \in \mathbb{N}, i \in [N]}$. It follows from the assumption on $\|d^k\|$ and step 2.4.a of Algorithm 2 that

$$\|z^k - \mathbf{u}^k\| \leq (1 - \tau_k)\|z^k - v^k\| + \tau_k\|d^k\| \leq (1 - \tau_k + \tau_k D)\|z^k - v^k\| \leq \eta_1 \|\bar{z}_N^{k-1} - z^k\|, \quad (\text{B.2})$$

where $\eta_1 = \bar{L}(1 - \tau_k + \tau_k D)$ and Lipschitz continuity of the proximal mapping was used in the last inequality. Further using triangular inequality yields

$$\begin{aligned}
\|\mathbf{u}^k - \bar{z}_N^{k-1}\| &\leq \|\mathbf{u}^k - z^k\| + \|z^k - \bar{z}_N^{k-1}\| & (\text{B.3}) \\
&\stackrel{(\text{B.2})}{\leq} (\eta_1 + 1) \|z^k - \bar{z}_N^{k-1}\|, \quad \text{and} \\
\|\tilde{z}_1^k - \mathbf{u}^k\| &= \frac{1}{\sqrt{N}} \|\tilde{z}_1^k - \mathbf{u}^k\| \\
&\leq \frac{1}{\sqrt{N}} \|\tilde{z}_1^k - z^k\| + \frac{1}{\sqrt{N}} \|z^k - \mathbf{u}^k\| \\
\text{Lip. continuity of } \text{prox}_{\Phi}^{\hat{H}} \text{ and Algorithm 2} &\leq \frac{\bar{L}}{\sqrt{N}} \|\mathbf{u}^k - \bar{z}_N^{k-1}\| + \frac{1}{\sqrt{N}} \|z^k - \mathbf{u}^k\| \\
(\text{B.3}), (\text{B.2}) &\leq \frac{1}{\sqrt{N}} ((\bar{L} + 1)\eta_1 + \bar{L}) \|z^k - \bar{z}_N^{k-1}\|.
\end{aligned}$$

Using this along with triangular inequality yields

$$\|\bar{z}_N^k - \mathbf{u}^k\| = \sum_{\ell=1}^N \|\tilde{z}_\ell^k - \mathbf{u}^k\| \stackrel{(\text{B.1})}{\leq} \sum_{\ell=1}^N c_\ell \|\tilde{z}_1^k - \mathbf{u}^k\| \leq \eta_2 \|z^k - \bar{z}_N^{k-1}\|,$$

where $\eta_2 = \sum_{\ell=1}^N \frac{c_\ell}{\sqrt{N}} ((\bar{L} + 1)\eta_1 + \bar{L})$. This inequality combined with (B.3) yields

$$\begin{aligned}
\|z^{k+1} - z^k\| &= \frac{1}{\sqrt{N}} \|\text{prox}_{\Phi}^{\hat{H}}(\bar{z}_N^k) - \text{prox}_{\Phi}^{\hat{H}}(\bar{z}_N^{k-1})\| \leq \frac{\bar{L}}{\sqrt{N}} \|\bar{z}_N^k - \bar{z}_N^{k-1}\| \\
&\leq \frac{\bar{L}}{\sqrt{N}} \|\bar{z}_N^k - \mathbf{u}^k\| + \frac{\bar{L}}{\sqrt{N}} \|\mathbf{u}^k - \bar{z}_N^{k-1}\| \leq \frac{\bar{L}}{\sqrt{N}} (\eta_1 + \eta_2 + 1) \|z^k - \bar{z}_N^{k-1}\|.
\end{aligned}$$

The claimed inequality follows from Lipschitz continuity of $\text{prox}_{\Phi}^{\hat{H}}$ and the inclusion in step 2.1 of Algorithm 2. \square

C Omitted proofs

Proof of Theorem 4.16

By level boundedness of φ and Theorem 4.7, $(\mathbf{u}^k)_{k \in \mathbb{N}}$, $(\mathbf{z}^k)_{k \in \mathbb{N}}$, $(z_i^k)_{k \in \mathbb{N}}$ are contained in a nonempty convex bounded set \mathcal{U} , where owing to Assumption 2.A2, h_i and consequently \hat{H} are strongly convex. It then follows from Fact A.1(ii), Theorem 4.7(ii), and Lemma B.2 that $\|z^{k+1} - z^k\| \rightarrow 0$. Therefore, the set of limit points of $(z^k)_{k \in \mathbb{N}}$ is nonempty compact and connected [11, Rem. 5]. By Theorems 4.7(iv) and 4.7(v) the limit points are stationary for φ , and $\Phi^{\hat{H}}(\mathbf{z}^k) = \mathcal{L}(v^k, z^k) \rightarrow \varphi_*$. In the trivial case $\Phi^{\hat{H}}(\mathbf{z}^k) = \mathcal{L}(v^k, z^k) = \varphi_*$ for some k , the claims follow from Theorem 4.7. Assume that $\Phi^{\hat{H}}(\mathbf{z}^k) > \varphi_*$ for $k \in \mathbb{N}$. The KL property for Φ is implied by that of φ due to Fact A.4, with desingularizing function $\psi(s) = \rho s^{1-\theta}$ with exponent $\theta \in (0, 1)$. Let Ω denote the set of limit points of $(z^k = (z^k, \dots, z^k))_{k \in \mathbb{N}}$. Since \hat{H} is strongly convex, [72, Lem. 5.1] can be invoked to infer that the function $\mathcal{M}_{\hat{H}}(\mathbf{w}, \mathbf{x}) = \Phi(\mathbf{w}) + D_{\hat{H}}(\mathbf{w}, \mathbf{x})$ also has the KL property with exponent $\nu \in \max\{\theta, \frac{1}{2}\}$ at every point $(\mathbf{z}^*, \mathbf{z}^*)$ in the compact set $\Omega \times \Omega$. Moreover, by (4.2) $\mathcal{M}_{\hat{H}}(\mathbf{z}^*, \mathbf{z}^*) = \Phi(\mathbf{z}^*) = \varphi_*$ where Theorem 4.7(iv) was used in the last equality. Recall that $\mathbf{z}^k \in \text{prox}_{\Phi}^{\hat{H}}(\bar{\mathbf{z}}_N^{k-1})$ as in step 2.1 of Algorithm 2. Therefore, $\partial \mathcal{M}_{\hat{H}}(\mathbf{z}^k, \bar{\mathbf{z}}_N^{k-1}) = \underbrace{(\partial \Phi(\mathbf{z}^k) + \nabla \hat{H}(\mathbf{z}^k) - \nabla \hat{H}(\bar{\mathbf{z}}_N^{k-1}), \nabla^2 \hat{H}(\bar{\mathbf{z}}_N^{k-1})(\bar{\mathbf{z}}_N^{k-1} - \mathbf{z}^k))}_{\geq 0, \text{ by (2.5) in the lifted space}}$, resulting in

$$\text{dist}(0, \partial \mathcal{M}_{\hat{H}}(\mathbf{z}^k, \bar{\mathbf{z}}_N^{k-1})) \leq \|\nabla^2 \hat{H}(\bar{\mathbf{z}}_N^{k-1})\| \|\bar{\mathbf{z}}_N^{k-1} - \mathbf{z}^k\| \leq c \|\bar{\mathbf{z}}_N^{k-1} - \mathbf{z}^k\| \quad (\text{C.1})$$

where $c = \sup_k \|\nabla^2 \hat{H}(\bar{\mathbf{z}}_N^{k-1})\| > 0$ is finite due to $\bar{\mathbf{z}}_N^k$ being bounded (cf. Theorem 4.7(vi)) and continuity of $\nabla^2 \hat{H}$. Considering (4.18) with (C.1), since $\mathcal{M}_{\hat{H}}(\mathbf{z}^k, \bar{\mathbf{z}}_N^{k-1}) = \Phi^{\hat{H}}(\bar{\mathbf{z}}_N^{k-1}) \rightarrow \varphi$ from above, and that $(\mathbf{z}^k, \bar{\mathbf{z}}_N^{k-1})_{k \in \mathbb{N}}$ is bounded and accumulates on $\Omega \times \Omega$, up to discarding iterates the following holds

$$\psi'(\Phi^{\hat{H}}(\bar{\mathbf{z}}_N^{k-1}) - \varphi_*) = \psi'(\mathcal{M}_{\hat{H}}(\mathbf{z}^k, \bar{\mathbf{z}}_N^{k-1}) - \mathcal{M}_{\hat{H}}(\mathbf{z}^*, \mathbf{z}^*)) \geq \frac{1}{c \|\bar{\mathbf{z}}_N^{k-1} - \mathbf{z}^k\|}, \quad (\text{C.2})$$

where $\psi = \rho s^{1-\nu}$ is a desingularizing function for $\mathcal{M}_{\hat{H}}$ on $\Omega \times \Omega$. Let us define

$$\Delta_k := \psi(\Phi^{\hat{H}}(\bar{\mathbf{z}}_N^{k-1}) - \varphi_*) = \rho[\Phi^{\hat{H}}(\bar{\mathbf{z}}_N^{k-1}) - \varphi_*]^{1-\nu} \leq \rho[1-\nu]c \|\bar{\mathbf{z}}_N^{k-1} - \mathbf{z}^k\|^{\frac{1-\nu}{\nu}}. \quad (\text{C.3})$$

Then, $\Delta_k^{\frac{\nu}{1-\nu}} \leq \rho \frac{1}{1-\nu} (1-\nu) \|\bar{\mathbf{z}}_N^{k-1} - \mathbf{z}^k\|$. Concavity of ψ also implies

$$\Delta_k - \Delta_{k+1} \geq \psi'(\Phi^{\hat{H}}(\bar{\mathbf{z}}_N^{k-1}) - \varphi_*)(\Phi^{\hat{H}}(\bar{\mathbf{z}}_N^{k-1}) - \Phi^{\hat{H}}(\bar{\mathbf{z}}_N^k)) \stackrel{\text{C.2}}{\geq} \frac{\Phi^{\hat{H}}(\bar{\mathbf{z}}_N^{k-1}) - \Phi^{\hat{H}}(\bar{\mathbf{z}}_N^k)}{c \|\bar{\mathbf{z}}_N^{k-1} - \mathbf{z}^k\|}. \quad (\text{C.4})$$

On the other hand by (4.11) and (4.10)

$$\Phi^{\hat{H}}(\bar{\mathbf{z}}_N^{k+1}) - \Phi^{\hat{H}}(\bar{\mathbf{z}}_N^k) \leq -D_{\hat{H}}(\mathbf{z}^{k+1}, \bar{\mathbf{z}}_N^k) \leq -\frac{\mu_{\hat{H}}}{2} \|\mathbf{z}^{k+1} - \bar{\mathbf{z}}_N^k\|^2, \quad (\text{C.5})$$

where Fact A.1(ii) was used and $\mu_{\hat{H}}$ denotes its strong convexity modulus. Combining (C.4) and (C.5),

$$\Delta_k - \Delta_{k+1} \geq \eta \|\bar{\mathbf{z}}_N^{k-1} - \mathbf{z}^k\| \geq \frac{\eta}{c} \|\mathbf{z}^{k+1} - \mathbf{z}^k\|, \quad (\text{C.6})$$

with some constant $\eta > 0$ where the last inequality follows from Lemma B.2. Hence, $(\|\mathbf{z}^{k+1} - \mathbf{z}^k\|)_{k \in \mathbb{N}}$ has finite length and is thus convergent. It then follows from Theorem 4.7(v) that $(\mathbf{z}^k)_{k \in \mathbb{N}}$ converges to a stationary point of φ . Combining (C.3) and (C.6) we have,

$$\Delta_{k+1} \leq \Delta_k - \alpha \Delta_k^{\frac{\nu}{1-\nu}} \quad (\text{C.7})$$

with some appropriate $\alpha \geq 0$. Hence, if $\nu = \frac{1}{2}$, i.e. $\theta \in (0, \frac{1}{2}]$ for Φ , in (C.7) we have $\Delta_{k+1} \leq (1-\alpha)\Delta_k$. As $\alpha > 0$ and $\frac{\Delta_{k+1}}{\Delta_k} > 0$, then $(1-\alpha) \in (0, 1)$ concluding Δ_k is Q-linearly convergent to zero. By (C.3) we then conclude $(\Phi^{\hat{H}}(\bar{\mathbf{z}}_N^{k-1}))_{k \in \mathbb{N}}$ is convergent Q-linearly and by Fact A.2(iii), where we have $\varphi(\mathbf{z}^k) = \Phi(\mathbf{z}^k) \leq \Phi^{\hat{H}}(\bar{\mathbf{z}}_N^{k-1})$, we conclude $(\varphi(\mathbf{z}^k))_{k \in \mathbb{N}}$ is convergent R-linearly. Moreover, the inequality (C.6) implies that $(\|\mathbf{z}^{k+1} - \mathbf{z}^k\|)_{k \in \mathbb{N}}$ is R-linearly convergent, thus so is $(\mathbf{z}^k)_{k \in \mathbb{N}}$.

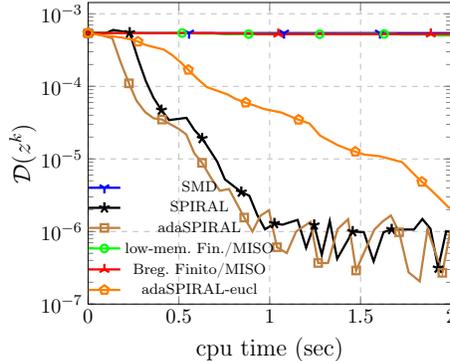


Figure 4: Performance of different algorithms versus cpu time on the phase retrieval problem (5.2) for 550 epochs on a digit 6 image with $N = 1280$, $n = 256$.

D CPU time

The performance results presented in Section 5 are also reported versus CPU time. According to the numerical comparisons in Figures 4 to 6, the proposed algorithm features relatively cheap iterations and has comparable computational complexity per epoch compared to the other algorithms.

E Algorithm variants

E.1 Adaptive variant

In this section, the implementation of Table 1 is further discussed. In Table 1, for the first iterate, i.e. $k = 0$, the vectors \tilde{z}_i^{-1} are initially considered equal to z^{init} for all $i \in [N]$. Also, note that the linesearch in step 2.5.d of Table 1 backtracks to step 2.3.a, rather than step 2.5.c. Performing the linesearches in this intertwined fashion is observed to result in acceptance of good directions and reduction in the overall computational complexity [20, 52]. We refer the reader to [20] for the theoretical justification for the effectiveness of this procedure. Note that in Algorithm 3, in the Euclidean case, the same backtracking can be used with dgfs $h_i = \frac{1}{2} \|\cdot\|^2$. The backtracking linesearches in the first block of Table 1 do not require storing \tilde{z}_i^k and can be performed efficiently. In step 2.1.b $\sum_{i=1}^N p_i(\cdot, \tilde{z}_i^k)$ may be evaluated by storing the scalars $\sum_{i=1}^N f_i(\tilde{z}_i^k)$ and $\sum_{i=1}^N \langle \nabla f_i(\tilde{z}_i^k), \tilde{z}_i^k \rangle$ and one vector $\sum_{i=1}^N \nabla f_i(\tilde{z}_i^k) \in \mathbb{R}^n$ while performing step 1.10 of the algorithm. Similar tricks apply to the computation of the Bregman distances, functions p_i in other backtracking linesearches of Table 1, and updating the vectors s^k , \bar{s}^k , and \tilde{s}^k .

E.2 Euclidean variant

In this section, the proposed algorithm in the Euclidean version is outlined in Algorithm 3, when the functions f_i have Lipschitz continuous gradients with constants L_i . In this case, the distance generating functions are $h_i = \frac{1}{2} \|\cdot\|^2$ and consequently the Bregman distances are simplified to $D_{h_i}(y, x) = \frac{1}{2} \|y - x\|^2$.

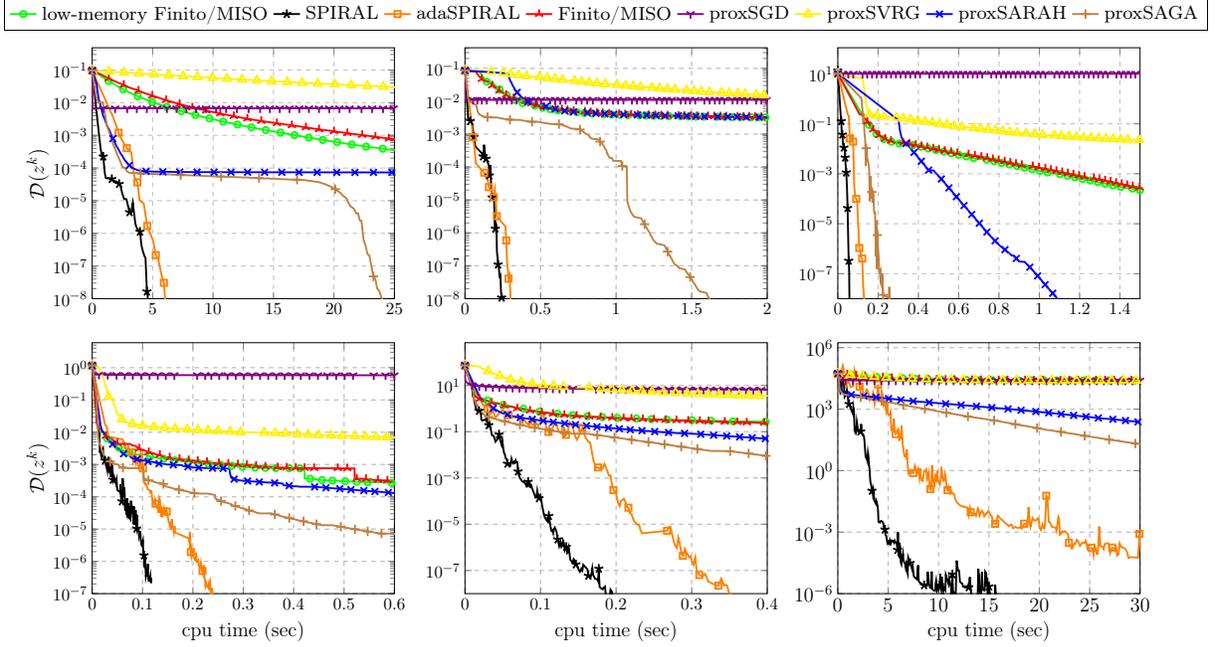


Figure 5: Performance of different algorithms versus cpu time on the lasso problem of (5.3) for 50 epochs. Synthetic dataset (top left) with $N = 10000$, $n = 400$, synthetic dataset (top center) with $N = 300$, $n = 600$, mg (top right) with $N = 1385$, $n = 6$, triazines (bottom left) with $N = 186$, $n = 60$, housing (bottom center) with $N = 506$, $n = 13$, and cadata (bottom right) with $N = 20640$, $n = 8$.

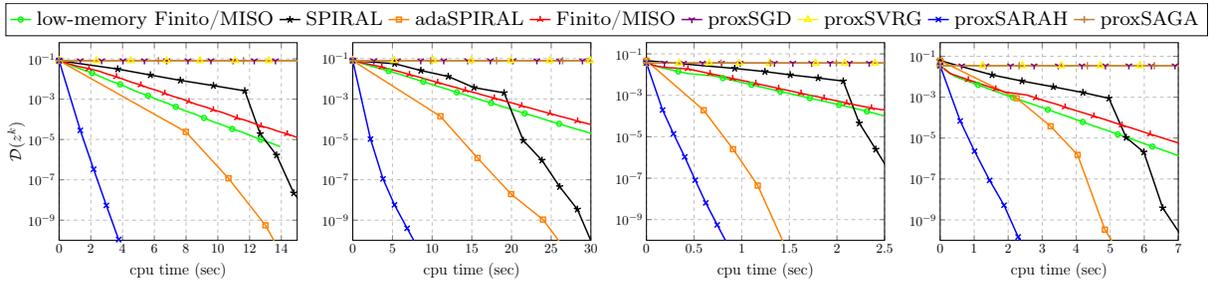


Figure 6: Performance of different algorithms versus cpu time on the NN-PCA problem of (5.4) for 500 epochs. MNIST (left) with $N = 60000$, $n = 784$, covtype (left center) with $N = 581012$, $n = 54$, a9a (right center) with $N = 32561$, $n = 123$, and aloi (right) with $N = 108000$, $n = 128$.

Algorithm 3 SPIRAL - Euclidean version

Require $z^{\text{init}} \in \mathbb{R}^n$, $\Gamma = \{\gamma_1, \dots, \gamma_N\}$ with $\gamma_i \in (0, N/L_i)$, $i \in [N]$, $\beta \in (0, 1)$,
 $\hat{\gamma} := (\sum_{i=1}^N \gamma_i^{-1})^{-1}$, $s^0 = z^{\text{init}} - \frac{\hat{\gamma}}{N} \sum_{i=1}^N \nabla f_i(z^{\text{init}})$,
 maximum number of backtracks $q_{\max} \in \mathbb{N} \cup \{\infty\}$ (e.g. $q_{\max} = 2$), $K \in \mathbb{N}$

Repeat for $k = 0, 1, \dots, K$

3.1: $z^k \in \text{prox}_{\hat{\gamma}g}(s^k)$

3.2: $\bar{s}^k = z^k - \frac{\hat{\gamma}}{N} \sum_{i=1}^N \nabla f_i(z^k)$ (full update)

3.3: $v^k \in \text{prox}_{\hat{\gamma}g}(\bar{s}^k)$

3.4: choose $d^k \in \mathbb{R}^n$ at z^k (e.g. based on a quasi-Newton method for solving $r_{\hat{h}}(z) = 0$)

3.5: set $\tau_k = 1, q_k = 0$ (linesearch)

a: $u^k = \tau_k z^k + (1 - \tau_k)v^k + \tau_k d^k$

b: $\tilde{s}^k = u^k - \frac{\hat{\gamma}}{N} \sum_{i=1}^N \nabla f_i(u^k)$ (full update)

c: $y^k \in \text{prox}_{\hat{\gamma}g}(\tilde{s}^k)$

d: **if** $\mathcal{L}(y^k, u^k) \leq \mathcal{L}(v^k, z^k)$
 go to step 3.6

e: **else if** $q_k = q_{\max}$ **then**

$u^k = v^k$, $\tilde{s}^k = u^k - \frac{\hat{\gamma}}{N} \sum_{i=1}^N \nabla f_i(u^k)$, and go to step 3.6

f: **else**

$\tau_k \leftarrow \beta \tau_k$, $q_k \leftarrow q_k + 1$, and go to step 3.5.a

3.6: $s^k \leftarrow \tilde{s}^k$

3.7: **for** $\ell = 1, \dots, N$ **do** (incremental loop)

3.8: randomly choose $i^\ell \in [N]$ without replacement

3.9: $\tilde{z}_{i^\ell}^k \in \text{prox}_{\hat{\gamma}g}(s^k)$

3.10: $s^k \leftarrow s^k + \frac{\hat{\gamma}}{N} (\nabla f_{i^\ell}(u^k) - \nabla f_{i^\ell}(\tilde{z}_{i^\ell}^k)) + \frac{\hat{\gamma}}{\gamma_{i^\ell}} (\tilde{z}_{i^\ell}^k - u^k)$

3.11: $s^{k+1} \leftarrow s^k$

Return z^K

Table 1: Adaptive stepsize selection for SPIRAL. Let $\alpha \in (0, 1)$, $\sigma \in (0, 1)$, and β as in Algorithm 1. Initialize the stepsizes γ_i for $i \in [N]$. For the sake of readability, define $p_i(y, x) := f_i(y) - f_i(x) - \langle \nabla f_i(x), y - x \rangle$.

2.1:

- a: $z^k \in t(s^k)$
 - b: **if** $\sum_{i=1}^N p_i(z^k, z_i^{k-1}) > \sum_{i=1}^N \frac{\alpha N}{\gamma_i} D_{h_i}(z^k, z_i^{k-1})$ **then**
 $\gamma_i \leftarrow \sigma \gamma_i$ for all $i \in [N]$, update s^k , and go to step 2.1.a
-

2.3:

- a: $v^k \in t(\bar{s}^k)$
 - b: **if** $\sum_{i=1}^N p_i(v^k, z^k) > \sum_{i=1}^N \frac{N}{\gamma_i} D_{h_i}(v^k, z^k)$ **then**
 $\gamma_i \leftarrow \sigma \gamma_i$ for all $i \in [N]$, update \bar{s}^k , and go to step 2.3.a
-

2.4: choose $d^k \in \mathbb{R}^n$ at z^k (e.g. based on a quasi-Newton method for solving $r_{\hat{h}}(z) = 0$)

2.5: set $\tau_k = 1, q_k = 0$

- a: $u^k = z^k + (1 - \tau_k)(v^k - z^k) + \tau_k d^k$
 - b: $\tilde{s}^k = \sum_{i=1}^N \frac{1}{\gamma_i} \nabla h_i(u^k) - \frac{1}{N} \nabla f_i(u^k)$
 - c: $y^k \in t(\tilde{s}^k)$
 - d: **if** $\sum_{i=1}^N p_i(y^k, u^k) > \sum_{i=1}^N \frac{N}{\gamma_i} D_{h_i}(y^k, u^k)$ **then**
 $\gamma_i \leftarrow \sigma \gamma_i$ for all $i \in [N]$, update \bar{s}^k , and go to step 2.3.a
 - e: **if** $\mathcal{L}(y^k, u^k) \leq \mathcal{L}(v^k, z^k)$
go to step 2.6
 - f: **else if** $q_k = q_{\max}$ **then**
 $u^k = v^k$ and go to step 2.5.b
 - g: **else**
 $\tau_k \leftarrow \beta \tau_k, q_k \leftarrow q_k + 1$, and go to step 2.5.a
-

2.6: $s^k \leftarrow \tilde{s}^k$

2.9:

- a: $\tilde{z}_{i^\ell}^k \in t(s^k)$
 - b: **if** $p_{i^\ell}(\tilde{z}_{i^\ell}^k, u^k) > \frac{N}{\gamma_{i^\ell}} D_{h_{i^\ell}}(\tilde{z}_{i^\ell}^k, u^k)$ **then**
 $\gamma_{i^\ell} \leftarrow \sigma \gamma_{i^\ell}$, update s^k , and go to step 2.9.a
-

References

- [1] Masoud Ahookhosh, Andreas Themelis, and Panagiotis Patrinos. A Bregman forward-backward linesearch algorithm for nonconvex composite optimization: Superlinear convergence to nonisolated local minima. *SIAM Journal on Optimization*, 31(1):653–685, 2021.
- [2] Francisco Javier Aragón Artacho, Anton Belyakov, Asen L Dontchev, and Marco López. Local convergence of quasi-newton methods under metric regularity. *Computational Optimization and Applications*, 58(1):225–247, 2014.
- [3] Heinz H Bauschke, Jérôme Bolte, and Marc Teboulle. A descent lemma beyond Lipschitz gradient continuity: first-order methods revisited and applications. *Mathematics of Operations Research*, 42(2):330–348, 2017.
- [4] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.
- [5] Yoshua Bengio. Practical recommendations for gradient-based training of deep architectures. *Neural Networks: Tricks of the Trade: Second Edition*, pages 437–478, 2012.
- [6] Dimitri P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 2016.
- [7] Dimitri P Bertsekas and John N Tsitsiklis. Gradient convergence in gradient methods with errors. *SIAM Journal on Optimization*, 10(3):627–642, 2000.
- [8] Doron Blatt, Alfred O Hero, and Hillel Gauchman. A convergent incremental gradient method with a constant step size. *SIAM Journal on Optimization*, 18(1):29–51, 2007.
- [9] Jérôme Bolte, Aris Daniilidis, and Adrian Lewis. The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM Journal on Optimization*, 17(4):1205–1223, 2007.
- [10] Jérôme Bolte, Aris Daniilidis, Adrian Lewis, and Masahiro Shiota. Clarke subgradients of stratifiable functions. *SIAM Journal on Optimization*, 18(2):556–572, 2007.
- [11] Jérôme Bolte, Shoham Sabach, and Marc Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1-2):459–494, 2014.
- [12] Jérôme Bolte, Shoham Sabach, Marc Teboulle, and Yakov Vaisbourd. First order methods beyond convexity and lipschitz gradient continuity with applications to quadratic inverse problems. *SIAM Journal on Optimization*, 28(3):2131–2151, 2018.
- [13] Xufeng Cai, Cheuk Yin Lin, and Jelena Diakonikolas. Empirical risk minimization with shuffled sgd: A primal-dual perspective and improved bounds. *arXiv preprint arXiv:2306.12498*, 2023.
- [14] Xufeng Cai, Chaobing Song, Stephen Wright, and Jelena Diakonikolas. Cyclic block coordinate descent with variance reduction for composite nonconvex optimization. In *International Conference on Machine Learning*, pages 3469–3494. PMLR, 2023.
- [15] Emmanuel J Candes, Xiaodong Li, and Mahdi Soltanolkotabi. Phase retrieval via Wirtinger flow: Theory and algorithms. *IEEE Transactions on Information Theory*, 61(4):1985–2007, 2015.
- [16] Jaeyoung Cha, Jaewook Lee, and Chulhee Yun. Tighter lower bounds for shuffling SGD: Random permutations and beyond. In *International Conference on Machine Learning*, pages 3855–3912. PMLR, 23–29 Jul 2023.

- [17] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2:1–27, 2011.
- [18] Gong Chen and Marc Teboulle. Convergence analysis of a proximal-like minimization algorithm using Bregman functions. *SIAM Journal on Optimization*, 3(3):538–543, 1993.
- [19] Damek Davis, Dmitriy Drusvyatskiy, and Kellie J MacPhee. Stochastic model-based minimization under high-order growth. *arXiv preprint arXiv:1807.00255*, 2018.
- [20] Alberto De Marchi and Andreas Themelis. Proximal gradient algorithms under local lipschitz gradient continuity: A convergence and robustness analysis of panoc. *Journal of Optimization Theory and Applications*, 194(3):771–794, 2022.
- [21] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in neural information processing systems*, pages 1646–1654, 2014.
- [22] Aaron Defazio and Justin Domke. Finito: A faster, permutable incremental gradient method for big data problems. In *International Conference on Machine Learning*, pages 1125–1133, 2014.
- [23] John E Dennis and Jorge J Moré. A characterization of superlinear convergence and its application to quasi-newton methods. *Mathematics of computation*, 28(126):549–560, 1974.
- [24] John E Dennis, Jr and Jorge J Moré. Quasi-newton methods, motivation and theory. *SIAM review*, 19(1):46–89, 1977.
- [25] Radu Alexandru Dragomir, Mathieu Even, and Hadrien Hendrikx. Fast stochastic bregman gradient methods: Sharp analysis and variance reduction. In *International Conference on Machine Learning*, pages 2815–2825. PMLR, 2021.
- [26] John C Duchi and Feng Ruan. Solving (most) of a set of quadratic equalities: Composite optimization for robust phase retrieval. *Information and Inference: A Journal of the IMA*, 8(3):471–529, 2019.
- [27] Francisco Facchinei and Jong-Shi Pang. *Finite-dimensional variational inequalities and complementarity problems*, volume II. Springer, 2003.
- [28] Cong Fang, Chris Junchi Li, Zhouchen Lin, and Tong Zhang. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. *Advances in neural information processing systems*, 31, 2018.
- [29] Rong Ge, Zhize Li, Weiyao Wang, and Xiang Wang. Stabilized svrg: Simple variance reduction for nonconvex optimization. In *Conference on learning theory*, pages 1394–1448. PMLR, 2019.
- [30] Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.
- [31] Saeed Ghadimi, Guanghui Lan, and Hongchao Zhang. Mini-batch stochastic approximation methods for nonconvex stochastic composite optimization. *Mathematical Programming*, 155(1-2):267–305, 2016.
- [32] Mert Gürbüzbalaban, Asu Ozdaglar, and Pablo A Parrilo. Why random reshuffling beats stochastic gradient descent. *Mathematical Programming*, 186:49–84, 2021.
- [33] Filip Hanzely and Peter Richtárik. Fastest rates for stochastic mirror descent methods. *Computational Optimization and Applications*, pages 1–50, 2021.

- [34] Jeff Haochen and Suvrit Sra. Random shuffling beats sgd after finite epochs. In *International Conference on Machine Learning*, pages 2624–2633. PMLR, 2019.
- [35] Trevor Hastie, Jerome Friedman, and Robert Tibshirani. *The Elements of Statistical Learning*. Springer New York, 2001.
- [36] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in neural information processing systems*, 26:315–323, 2013.
- [37] Chao Kan and Wen Song. The Moreau envelope function and proximal mapping in the sense of the Bregman distance. *Nonlinear Analysis: Theory, Methods & Applications*, 75(3):1385 – 1399, 2012.
- [38] Krzysztof Kurdyka. On gradients of functions definable in o -minimal structures. *Annales de l’institut Fourier*, 48(3):769–783, 1998.
- [39] Puya Latafat, Andreas Themelis, Masoud Ahookhosh, and Panagiotis Patrinos. Bregman finito/miso for nonconvex regularized finite sum minimization without lipschitz gradient continuity. *SIAM Journal on Optimization*, 32(3):2230–2262, 2022.
- [40] Puya Latafat, Andreas Themelis, and Panagiotis Patrinos. Block-coordinate and incremental aggregated proximal gradient methods for nonsmooth nonconvex problems. *Mathematical Programming*, pages 1–30, 2021.
- [41] Zhize Li and Peter Richtárik. ZeroSARAH: Efficient nonconvex finite-sum optimization with zero full gradient computation. *arXiv preprint arXiv:2103.01447*, 2021.
- [42] Haihao Lu, Robert M. Freund, and Yurii Nesterov. Relatively smooth convex optimization by first-order methods, and applications. *SIAM Journal on Optimization*, 28(1):333–354, 2018.
- [43] Julien Mairal. Incremental majorization-minimization optimization with application to large-scale machine learning. *SIAM Journal on Optimization*, 25(2):829–855, 2015.
- [44] Konstantin Mishchenko, Ahmed Khaled, and Peter Richtárik. Random reshuffling: Simple analysis with vast improvements. *Advances in Neural Information Processing Systems*, 33:17309–17320, 2020.
- [45] Aryan Mokhtari, Mark Eisen, and Alejandro Ribeiro. IQN: An incremental quasi-Newton method with local superlinear convergence rate. *SIAM Journal on Optimization*, 28(2):1670–1698, 2018.
- [46] Aryan Mokhtari, Mert Gürbüzbalaban, and Alejandro Ribeiro. Surpassing gradient descent provably: A cyclic incremental method with linear convergence rate. *SIAM Journal on Optimization*, 28(2):1420–1447, 2018.
- [47] Philipp Moritz, Robert Nishihara, and Michael Jordan. A linearly-convergent stochastic L-BFGS algorithm. In *Artificial Intelligence and Statistics*, pages 249–258. PMLR, 2016.
- [48] Angelia Nedic and Soomin Lee. On stochastic subgradient mirror-descent algorithm with weighted averaging. *SIAM Journal on Optimization*, 24(1):84–107, 2014.
- [49] Yu. Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140(1):125–161, aug 2013.
- [50] Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 137. Springer Science & Business Media, 2018.
- [51] Lam M Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. SARAH: A novel method for machine learning problems using stochastic recursive gradient. In *International Conference on Machine Learning*, pages 2613–2621. PMLR, 2017.

- [52] Pieter Pas, Mathijs Schuurmans, and Panagiotis Patrinos. Alpaqa: A matrix-free solver for nonlinear mpc and large-scale nonconvex optimization. In *2022 European Control Conference (ECC)*, pages 417–422. IEEE, 2022.
- [53] Nhan H Pham, Lam M Nguyen, Dzung T Phan, and Quoc Tran-Dinh. ProxSARAH: An efficient algorithmic framework for stochastic composite nonconvex optimization. *J. Mach. Learn. Res.*, 21:110–1, 2020.
- [54] Benjamin Recht and Christopher Ré. Parallel stochastic gradient algorithms for large-scale matrix completion. *Mathematical Programming Computation*, 5(2):201–226, 2013.
- [55] Sashank J. Reddi, Ahmed Hefny, Suvrit Sra, Barnabas Póczos, and Alexander J. Smola. Stochastic variance reduction for nonconvex optimization. In *International conference on machine learning*, pages 314–323, 2016.
- [56] Sashank J. Reddi, Suvrit Sra, Barnabas Póczos, and Alexander J. Smola. Proximal stochastic methods for nonsmooth nonconvex finite-sum optimization. In *Advances in Neural Information Processing Systems*, pages 1145–1153, 2016.
- [57] R. T. Rockafellar and R. J.-B. Wets. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.
- [58] Ralph Tyrell Rockafellar. *Convex analysis*. Princeton University Press, 1970.
- [59] Anton Rodomanov and Dmitry Kropotov. A superlinearly-convergent proximal Newton-type method for the optimization of finite sums. In *International Conference on Machine Learning*, pages 2597–2605. PMLR, 2016.
- [60] Hamed Sadeghi and Pontus Giselsson. Hybrid acceleration scheme for variance reduced stochastic optimization algorithms. *arXiv preprint arXiv:2111.06791*, 2021.
- [61] Mark Schmidt, Nicolas Le Roux, and Francis Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1):83–112, mar 2017.
- [62] Shai Shalev-Shwartz and Tong Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14(Feb):567–599, 2013.
- [63] Mikhail V. Solodov and Benar F. Svaiter. An inexact hybrid generalized proximal point algorithm and some new results on the theory of Bregman functions. *Math. Oper. Res.*, 25(2):214–230, 2000.
- [64] Ju Sun, Qing Qu, and John Wright. A geometric analysis of phase retrieval. *Foundations of Computational Mathematics*, 18(5):1131–1198, 2018.
- [65] Marc Teboulle. A simplified view of first order methods for optimization. *Mathematical Programming*, 170(1):67–96, 2018.
- [66] A. Themelis and P. Patrinos. Supermann: A superlinearly convergent algorithm for finding fixed points of nonexpansive operators. *IEEE Transactions on Automatic Control*, 64(12):4875–4890, dec 2019.
- [67] Andreas Themelis, Masoud Ahookhosh, and Panagiotis Patrinos. On the acceleration of forward-backward splitting via an inexact Newton method. In Heinz H. Bauschke, Regina S. Burachik, and D. Russell Luke, editors, *Splitting Algorithms, Modern Operator Theory, and Applications*, pages 363–412. Springer International Publishing, Cham, 2019.

- [68] Andreas Themelis, Lorenzo Stella, and Panagiotis Patrinos. Forward-backward envelope for the sum of two nonconvex functions: Further properties and nonmonotone linesearch algorithms. *SIAM Journal on Optimization*, 28(3):2274–2303, 2018.
- [69] N Denizcan Vanli, Mert Gurbuzbalaban, and Asuman Ozdaglar. Global convergence rate of proximal incremental aggregated gradient methods. *SIAM Journal on Optimization*, 28(2):1282–1300, 2018.
- [70] Zhe Wang, Kaiyi Ji, Yi Zhou, Yingbin Liang, and Vahid Tarokh. Spiderboost and momentum: Faster variance reduction algorithms. *Advances in Neural Information Processing Systems*, 32, 2019.
- [71] Minghan Yang, Andre Milzarek, Zaiwen Wen, and Tong Zhang. A stochastic extra-step quasi-Newton method for nonsmooth nonconvex optimization. *Mathematical Programming*, pages 1–47, 2021.
- [72] Peiran Yu, Guoyin Li, and Ting Kei Pong. Kurdyka-Łojasiewicz exponent via inf-projection. *Foundations of Computational Mathematics*, pages 1–47, 2021.
- [73] Hui Zhang, Yu-Hong Dai, Lei Guo, and Wei Peng. Proximal-like incremental aggregated gradient method with linear convergence under Bregman distance growth conditions. *Mathematics of Operations Research*, 46(1):61–81, 2021.
- [74] Jiaojiao Zhang, Huikang Liu, Anthony Man-Cho So, and Qing Ling. Variance-reduced stochastic quasi-Newton methods for decentralized learning: Part I, 2022.